HyperCube

HyperCube
Pictorial User's Guide

Contents

About HyperCube

HyperCube is a Macintosh and Windows application program specifically directed to the analysis and display of multi and hyperspectral imagery. This includes the static and dynamic display of the image cube and the generation of spectral classifications using both imagery and spectral libraries. In addition, **HyperCube** contains functions to filter, warp, mosaic, reformat, calibrate, combine, photogrammetrically project and to perform arithmetic on imagery and data.

The current version number of **HyperCube** is shown in the lower right hand corner of the splash screen by selecting About... in the Apple menu or the Help menu in the Windows version.

Macintosh Version

The Mac version of **HyperCube** is a "Carbonized" OS X native program that will also run on the Mac Classic OS, versions 8.6 and above. If running on the later OS use Get Info in the Finder to allocate as much memory as possible. **HyperCube** will run with only 8 MB but some functions may have to be broken into successive operations. More typically, 80 MB is a good size. Scaled overviews permit selectively loading full resolution subsets into memory. **HyperCube**'s Info window (described in Image Operations) will show a running tally of available memory.

Memory limitations and sizing are not applicable when running on Mac OS X as the OS will dynamically allocate memory as necessary.

Windows Version

The Windows version runs on the Windows® 95 through XP operating systems. There are no other special requirements or considerations. The Mac and Windows versions are 99.9% identical with one significant difference: the Windows version does not support the Mac Pict image format. It uses TIFF as the basic image format. Although this documentation was done on a Mac version of PageMaker® using Mac examples, the Windows GUI matches it one-for-one. Any slight (very slight) differences between the two versions is delineated in the text (in particular see the "Load Selection" button in section Band List). Also, see "Windows version" in the index.

The Windows version uses the Control (Ctrl) key instead of the Macintosh special command key. So, for example, to perform a paste operation press the Ctrl and V keys simultaneously. The Mac Option key is replaced with the Alt key in Windows.

Loading Images

Intrinsic File Types

The intrinsic image file format for the <u>Mac</u> is TIFF for OS X and Pict for the Classic OS. HyperCube also automatically recognizes Pict, TIFF, JPEG and .SUN (SRF) file types in the menu File->Open... (shown at the right) provided their Finder <u>file types</u> (not the file name suffix) are properly set. If the list does not show the file you are expecting then the file type isn't one of the above and the image must be opened using menu File->Open As... with you specifying the file type. If Edit->Options->Detailed Messages is checked then the header contents of the various file types will also be listed and is a good diagnostic tool.
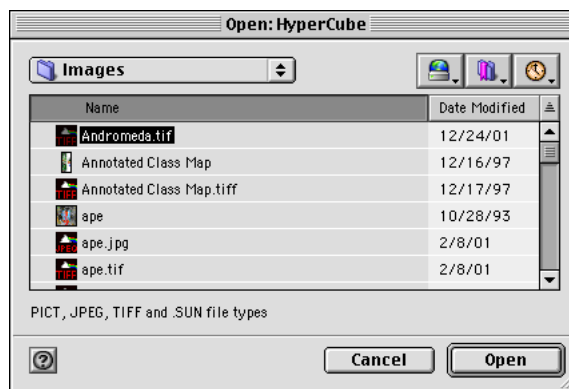
The <u>Windows</u> version of HyperCube uses File->Open... (shown below) to load all types of image formats except Pict which is not supported. TIFF is the general file type for <u>Windows</u>. Use the Files of type: popup menu to select which type.
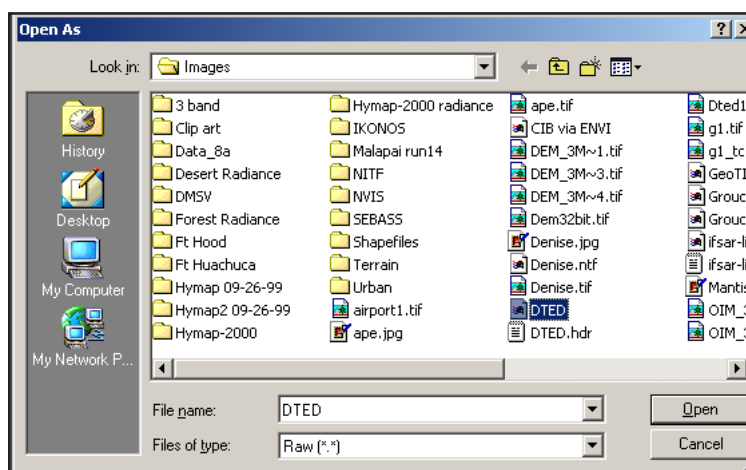
Windows standard open file dialog.

SRF (Sun Raster Files) and TARGA

Sun raster (SRF) and TARGA formatted images can be loaded by selecting Open As... from the File menu and then choosing the appropriate item in the Format popup menu. It is up to you to know which images are actually of that format. No other dialog is required as these images contain sizing information in their header. HyperCube should be able to read 8 bit and 24 bit compressed and uncompressed Sun images. HyperCube can also save images in the Sun format but not TARGA. The TARGA format is included for historical reasons.

ESRI Shape Files

HyperCube can load and display certain types of ESRI (Environmental Systems Research Institute, Inc.) Shape Files. In particular are those produced by HyperCube via the Class Map Editor Export function. Shape files consist of a main file that should have a .shp extension along with the index file (.shx extension) and a dbase file (.dbf extension). No use is made of the last two files but their contents can be displayed in a scrollable text window by selecting File->Open As..., choosing ESRI Shape and selecting the desired file extension type. For convenience, the_main shape file itself does not have to have the .shp extension whereas the other two do require their proper extension.

Shown below is the result of opening the three shape files that were exported by the Utilities->Class Map Editor. Each text window contains a parsing and because of the possible large size of shape files not all of the records may be shown. Also, the shape plot sizing is abitrary unless it was exported from HyperCube.
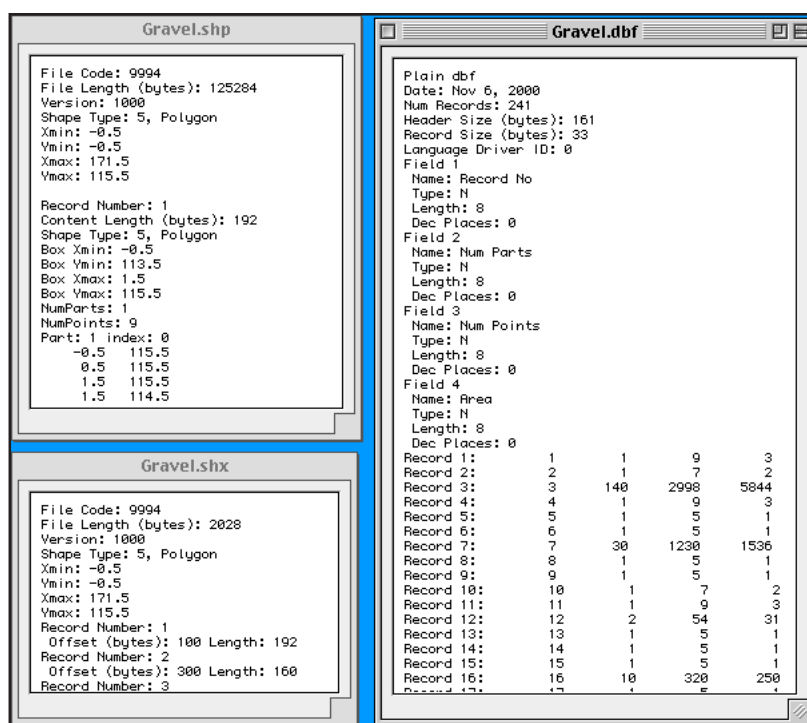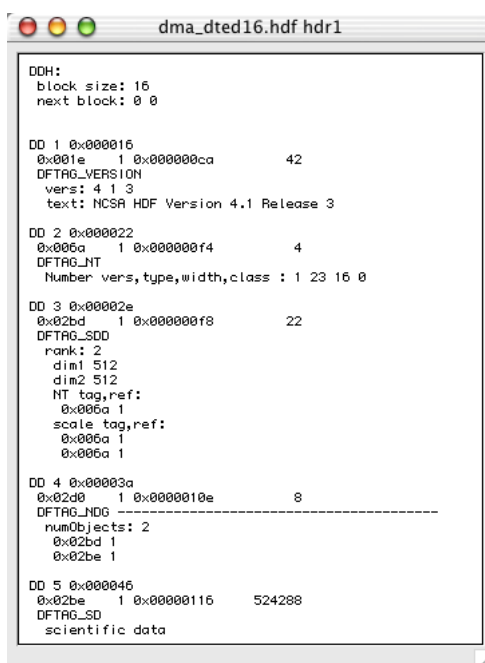
The three shape files and the corresponding plot that were produced via the Utilities->Class Map Editor Export *function. Only a portion of each file is shown in the text windows. See section* Class Map Editor *for more details.*
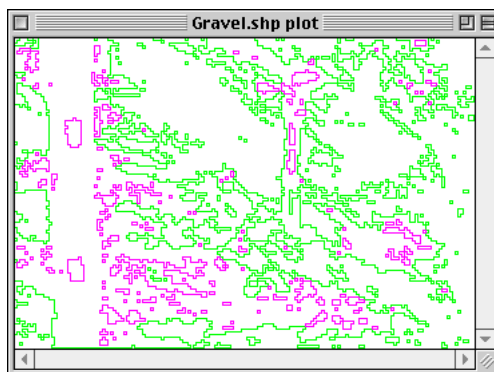
HDF Files

The Hierarchical Data Format (HDF) was developed at the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign. Most of the image/data formats described in the NCSA HDF Specification and Developer's Guide, Version 4.1r5, Novmenber 2001 can be loaded and saved in HyperCube. The strong exceptions are: the complex data type is not supported and the number of dimensions (rank) must be less than or equal to three for Numeric Data Groups. HyperCube's HDF capability is still in progress and there are other defined formats that may or may not work. In addition, HyperCube can not save an image cube in HDF only single plane images may be saved in that format.

Load an HDF image by selecting menu File -> Open As... and choosing HDF as the file type. Once loaded there is no relationship between the image and its HDF file. Unlike HyperCube's handling of NITF and TIFF (see the corresponding sections) there is presently no overview capability, the entire image(s) is loaded as is. If menu item Edit -> Options -> Detailed Messages is toggled on (checked) then a listing of all of the recognized tags in the header portion of the HDF file is generated. A typical header from a 512 by 512 by 16 bit image is shown below.

Baseline JPEG files may be loaded and saved by HyperCube. Specifically, these are defined by ISO 10918 as:

- DCT - base process
- 8 and 16 bit samples per component (a little beyond baseline)
- Sequential
- Huffman coding
- 1,2,3 or 4 components
- Interleaved and non-interleaved scans

To load a JPEG image file select File->Open As..., JPEG and choose the file. If the image is not in JPEG format then you will get an alert message. Once loaded, a JPEG image is treated like any other loaded image and can be saved in any of the image formats.

Saving an image, either gray, pseudo or true color, is done via File->Save As..., JPEG when the image is the active window. This displays a subsequent dialog (after the standard save file dialog) that allows you to choose the level of compression as shown below. Also shown on the right is a text file dump of the various JPEG markers for the loaded image which appears if Edit->Options->Detailed Messages has been checked before the image is loaded.

Note: Because JPEG is a lossy format you cannot save a classmap in JPEG format. The color correspondence will be lost, use NITF, PICT, SRF or TIFF.
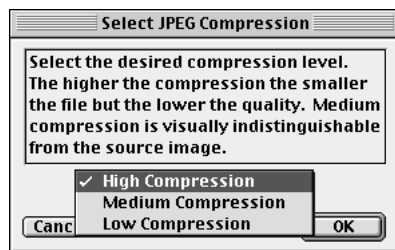
Three levels of compression available to you when saving an image in JPEG format.

A portion of the JPEG markers defining the contents a loaded JPEG image. This text window appears automatically if Edit->Preferences->Detailed Messages *is checked.*
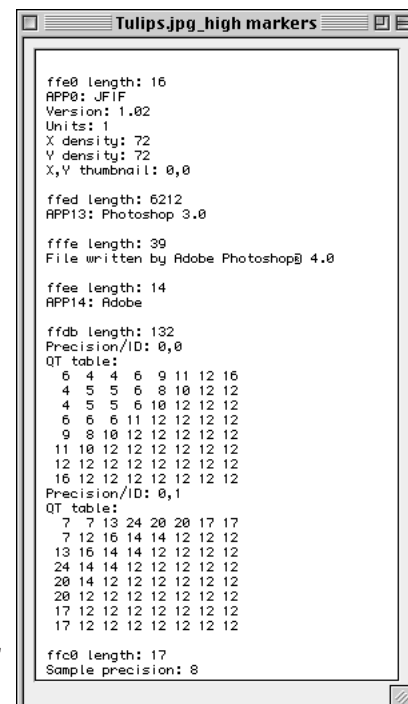
NITF Files

NITF (National Imagery Transmission Format) images are loaded via menu File->Open As... and then choosing the NITF item in the Format popup menu. HyperCube can read versions 2.0 and 2.1 NITF data, gray, pseudo, true color and multiband images including embedded baseline JPEG. Any graphics or text within the NITF file is ignored. If the image is multiband, larger than 1024 (pixels or lines) or the pixel depth is greater than 8 bits (e.g., short or float) you are presented with the dialog shown below. This allows you to load either an overview of the image (necessary if very large) or the entire image. Overview images are then used to select full resolution subsets that are subsequently loaded using menu File->Load Selection (see section Overviews). If there is insufficient memory to load then you will be alerted with a message stating how much additional memory is needed. The Retain source... option requires more memory depending on the pixel depth. Saving a 16 or 32 bit image as NITF saves the retained data as short or float. If Retain source... was not selected when the image was loaded then the 8 bit display values are saved.

If the NITF image is multiband you may select the starting band and the number of bands to load and if Overview is chosen, the overview band and scale as well. To exploit a NITF multiband image you must first save the loaded image cube. However, there are possible memory constraints doing this and the best technique is to use menu Utilities->Reformat Cube File to convert the NITF file into an image cube file.

A description of the image.

What part of the image to load. Initially, set to the image size.

If multiband which bands to load.

For the given size image, the scale automatically computed to display the whole image within the monitor.

How to remap the data for display.

Set this to keep the source file data in memory. This data will be used in place of the display data in all operations.
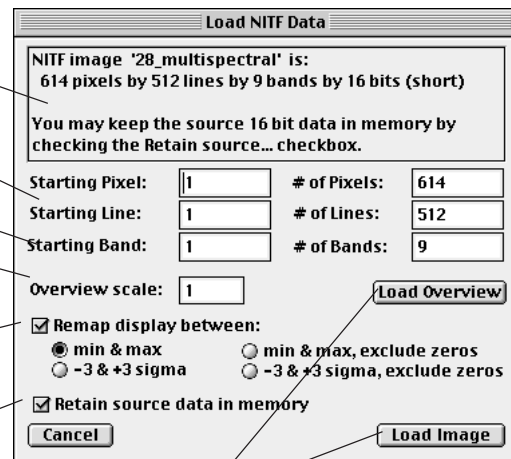
Select either an overview or the image at full resolution. You will be notified if insufficient memory.

If menu Edit->Options->Detailed Messages *is checked then each* NITF *image load will produce windows listing the main header and image header contents.*
Also, if the image header defined a coordinate system, use the Info Window *to display them.*
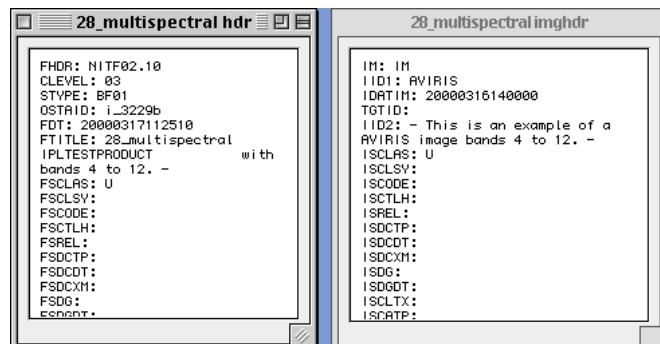
TIFF Files

TIFF (Tagged Image File Format) images are loaded via menu File->Open As... and then choosing the TIFF item in the Format popup menu. HyperCube can read 8 bit and higher baseline TIFF 6.0 images plus 16 and 32 bit tiled format images. Note that neither LZW nor embedded JPEG encoding are included in the baseline TIFF although JPEG images can be read by themselves (see section JPEG Files). If the image is larger than 1024 (pixels or lines) or the pixel depth is greater than 8 bits (e.g., short or float) you are presented with the dialog shown below. This allows you to load either an overview of the image (necessary if very large) or the entire image. Overview images are then used to select full resolution subsets that are subsequently loaded using menu File->Load Selection (see section Overviews). If there is insufficient memory to load then you will be alerted with a message stating how much additional memory is needed. The Retain source... option requires more memory depending on the pixel depth. Saving a 16 or 32 bit image as TIFF saves the retained data as short or float in strip format even if the source format was tiled. If Retain source... was not selected when the image was loaded then the 8 bit display values are saved.

If the TIFF image contains Geo tags (e.g., it's a GeoTIFF) then any coordinates will be extracted and used as the image reference (see section Image to Reference Coordinates). Saving a GeoTIFF embeds the four image corners as Geo tie points regardless of the source.

A description of the image.

What part of the image to load. Initially, set to the image size.

Not used for TIFF*, see* NITF*,*

For the given size image, the scale automatically computed to display the whole image within the monitor.

How to remap the data for display.

Set this to keep the source file data in memory. This data will be used in place of the display data in all operations.

Load TIFF Data

TIFF image 'Dem32bit.tif' is:
2227 pixels by 3095 lines by 32 bits (float)

You may keep the source 32 bit data in memory by checking the Retain source... checkbox.

Starting Pixel:	1	# of Pixels:	2227
Starting Line:	1	# of Lines:	3095
Starting Band:	N/A	# of Bands:	N/A

Overview scale: 6   [Load Overview]

☑ Remap display between:
　○ min & max　　　　　　○ min & max, exclude zeros
　◉ -3 & +3 sigma　　　　○ -3 & +3 sigma, exclude zeros

☑ Retain source data in memory

[Cancel]　　　　　　　　　　　　　[Load Image]
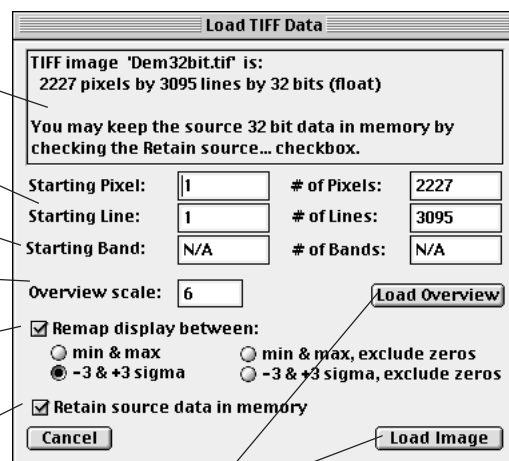
Select either an overview or the image at full resolution. You will be notified if insufficient memory.

If menu Edit->Options->Detailed Messages *is checked then each* TIFF *image load will produce the window at right listing the main* TIFF *tags and any Geo tags.*

Dem32bit.tif tags

```
Pixels, Lines, Bits:  2227, 3095, 32
SamplesPerPixel:  1
PlanarConfiguration: 1

RowsPerStrip:  3095
StripByteCounts[0,1,2,...]:
0,0,0,...
Number of strips: 0

Number of tiles: 450
Tile width: 128
Tile height: 128
TileByteCounts[0,1,2,...]:
65536,65536,65536,...
TileOffsets[0,1,2,...]:
8,65544,131080,...

Compression:  1
PhotometricInterpretation:  1
XResolution:  0
YResolution:  0
ResolutionUnit:
ByteOrder:  Big-endian
```

Raw

The Raw image format permits you to specify the dimensions and format of general image files and the resulting size of what will be loaded into an image window. The only difference between a Raw format and Multiband format (explained later) is that the Raw format results in loading a single image plane. Opening a Raw image always results in displaying an 8 bit image with the option of retaining the source data in memory. You have considerable control on how this file data is converted to 8 bits for display. The Raw image file to be opened is selected by choosing Open As... from the File menu and then selecting Raw from the Format popup menu. Since any file can be considered as a Raw image, the open dialog displays all file types. You must know which files represent meaningful images.

If option Retain source data values is checked (see dialog on following page) then the actual data is loaded as well. Then any operations that process the image (e.g. Fourier, Image Arithmetic, Line Profile Plots, Mosaic, Warp) will use the actual data instead of the resulting 8 bit display values. Moving the cursor over a data mage displays the data value when the Info window is open. Histograms and Line Profile Plot operate on the data. See the Applications section for examples of data usage.
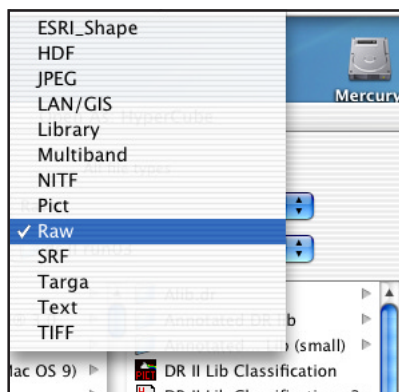
The Open As... *dialog showing the selection of a* Raw *format file.*

Multiband and LAN/GIS

A multiband image differs from a raw image in that it consists of 2 or more registered bands each having the same spatial dimensions. Multiband images are selected from the same custom open file dialog as Raw shown above but with Multiband chosen from the Format popup menu when you select Open As... from the File menu. Raw and multiband images are defined by a format dialog (see below) that appears immediately after the Open button is clicked. This dialog lists all of the information necessary to correctly read and interpret the image file and to load the image into a window for display. The initial content of the dialog is read from a simple ASCII text header file having the same name as the image file but with a .hdr extension (see section Header Files). If this file doesn't exist (i.e. has yet to be created) then the dialog fields are empty. In either case you can edit the dialog fields and save the proper values to create/update the *.hdr file. Subsequent opening of the image or image.hdr file will display the image definition in the format dialog.

If a multiband image is known to be in the LAN/GIS format with a properly embedded header then it may be loaded by selecting LAN/GIS from the Format popup menu. Hyper-Cube will attempt to decipher the header (either byte order) and preset these values into the dialog shown on the following page. It is then possible to edit and save the header (in Hyper-Cube format) for subsequent loading. The actual LAN/GIS header is not modified.

An image format dialog showing the definition of an 16 bit 512 by 512 raw image ready to load the entire image with remapping. The Pixel type *is unsigned short and* Byte order *indicates big endian. The image will be converted to 8 bits for display but will also retain the 16 bit underlying data that will be used in all operations.*
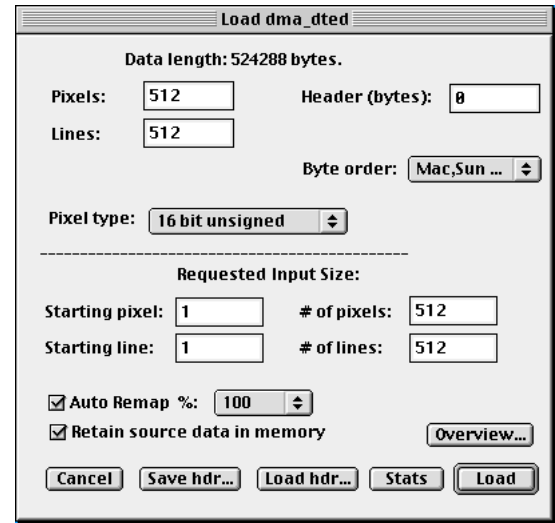
The upper portion of the dialog window defines the parameters of the image as it appears on disk. Only the data length is determined by HyperCube, all other information must come from you or the .hdr file. However, HyperCube does check the input for consistency.

Pixels: The number of pixels (columns) per line (row).

Lines: The number of lines (rows).

Header: The number of bytes, if any, to skip at the beginning of the file.

Pixel type: A popup menu specifying the format of each pixel
                8 bit unsigned
                16 bit unsigned
                16 bit signed
                32 bit floating
                32 bit signed long

Byte order: A popup that indicates whether the most significant data byte is low order in memory (little endian) or high order (big endian). Only meaningful for 16 bit and larger data. If gray values seem incorrect, try a different order.
                PC (little endian)
                Mac,Sun (big endian)
                Unknown.

The lower portion of the dialog window is where you specify what rectangular subset of the image file is to be loaded into memory. There are also options that determine how the pixels are to be converted to 8 bits for display. This isn't a concern with the example dialog shown since the data is also 8 bits deep. Pixel conversion is explained more fully in the later multiband example.

Specifically, the requested size fields have the following effect:

Starting pixel: The first pixel within a line to load and corresponds to the left edge of the resulting image display window. Pixel numbering begins with 1, not zero.

Starting line: The first line of the image file to start loading and corresponds to the top line in the image display window. Line numbering begins at 1.

# of pixels: How many pixels per line should be loaded.

# of lines: How many lines of the image are to be loaded.

Auto Remap: If this check box is selected then the image pixels are automatically converted from their pixel type to 8 bits based upon the distribution of their histogram and upon the percentage popup menu. Generally, this is meaningful only for 16 bit and wider data.

Retain source...   If checked then the source data is retained in memory at full depth. Retained data is used in computations instead of 8 bit display data.

%:   If Auto Remap is checked then this popup specifies what percentage of the distribution of the histogram about the median is mapped to the display interval [0,255]. See the statistical display window shown in the multiband dialog example.

Save hdr...   This button brings up the standard Mac save file dialog and allows you to save the *.hdr file defining this image. Subsequently, opening this image will refill the dialog fields from the *.hdr file.

Load hdr...   Displays an Open file dialog so that you can load a possibly different header file. There are no restrictions on the file's name.

Stats: When clicked, this button reads the image file and generates various statistics within the rectangular area that you want to load. The results are displayed graphically in a new window. See section Statistics.

Cancel: The customary button for aborting the image load.

Load: Loads the image automatically computing statistics as necessary.

Overview: This button loads a reduced resolution view of the source image. The initial scaling is determined by the display screen size and the Starting pixel, Starting line, # of pixels, and # of lines. You can select a different scaling. See section Overviews.

The image format dialog window for a <u>multiband</u> image is shown below. It includes all of the raw image fields plus additional ones to define band depth and the relationship of the file pixel values to an absolute [0,1] range for comparison with a reflectance library or when classifying one image cube against a different image cube (see section Classify Function).

The cube image format dialog showing the definition of a 210 band image with 16 bit pixels. Auto remapping of a 95% interval about the median will be used to load bands 60 through 69. The pixel values were previously calibrated to lie between 0 and 1000 will be converted to a [0,1] range by scale.

If Retain source... *is checked then the source pixel depth data is attached to the display image.*

```
┌──────────── Load FTH5_8 ────────────┐
        Data length: 172032000 bytes.

  Pixels:    [320]        Header: (bytes)  [0]
  Lines:     [1280]       Interleave:  [Line  ▾]
  Bands:     [210]        Byte order:  [Mac,Sun ... ▾]
  Pixel type:  [16 bit unsigned  ▾]
  Normalization to convert data to [0,1] interval:
     scale: [0.001000]   offset: [0.0000]
  -------------------------------------------------
              Requested Input Size:
  Starting pixel: [9]      # of pixels:  [307]
  Starting line:  [1]      # of lines:   [1280]
  Starting band:  [60]     # of bands:   [10]
  ☑ Remap Bands as a Group, %:  [95  ▾]
  ☐ Remap Bands Individually, sigma:  [2.5]
  ☑ Retain source data in memory
  ☐ Use G & O   [Load Gains & Offsets...]   [Overview...]
  [Cancel] [Save hdr...] [Load hdr...] [Stats] [Load]
```

The <u>additional</u> fields are defined as:

Bands:	Total number of bands in the image.
Interleave:	A popup menu indicating the type of data arrangement: Band Line 1 through N for band 1, Line 1 through N for band 2,... Line Line 1 for all bands, Line 2 for all bands,... Pixel Pixel 1 for all bands, Pixel 2 for all bands,...
scale & offset:	These are the transformation coefficients that convert a file pixel to the range [0,1]. This has nothing to do with display only spectral library matching and classifications with another cube. In the example above the file pixels lie between 0 and 1000 so the transformation is .001 and 0. Also, see Load Gains & Offsets.
Starting band:	Which band from the image will be the first display band. Band numbering begins at 1.
# of bands:	How many sequential bands beginning at the starting band should be loaded.
Load hdr...	Displays an Open file dialog so that you can load a possibly different header file. There are no restrictions on the file's name.

Remap Bands Individ: Each band's statistics is used to rescale it for presentation.
Choosing this option produces a display cube having a
uniform brightness and contrast. This only alters the display
and does not affect any subsequent analytical operations.
Also, see Remap Bands as a Group which preserves
the relative band to band brightness differences.

... Sigma: For each band's mean and standard deviation, pixels within
[mean - sigma*sd, mean + sigma*sd] are mapped to [0,255].

Remap Bands as a Group: The composite statistics for all of the bands that are
to be loaded are computed and applied globally. The result
is a display cube that represents the intensity variation of the
image cube file.

Load Gains & Offsets... Allows you to open a text file containing a list of gains and
offset pairs that will be applied every time cube file data is
accessed and affects all analytical operations such as
plots and classifications. The number of gain & offset pairs
must equal the number of cube file bands, not just the number
of bands loaded. After a gains & offsets has been loaded the
the data is internally converted to floating even if the source
file data is integer and the gains and offsets were also integers.
Therefore, subsequent operations will be a little slower than
if no gains and offsets were in effect. The scale and offset
fields that are used in Normalization to convert data to [0,1]
interval are applied after any gains & offsets, so be sure to
change this transformation to match the new range.
After loading a G & O file the Remap Bands check box will
be on. This is necessary due to the new floating point range.
Remap can be checked off but the resultant display image
cube will probably be very unsatisfactory. In any case, this
will not affect the plots or classifications.
See sections Spectral Calibration and Reformatting a Cube
File for more on gains & offsets.

Use G & O: Automatically enabled and checked on after a gains and offsets
file is loaded but may be checked off to discard the gains and
offsets. To determine later whether a G & O file is being used
by a cube open the Info window, (Windows->Show Info), and
examine the Kind field. It will show Kind: CUBE w/G&O if
one is being used.

The remaining cube file dialog fields are identical to those of the raw format.
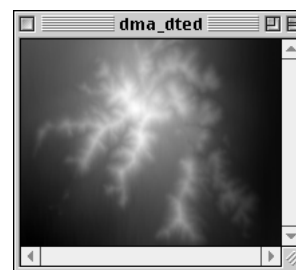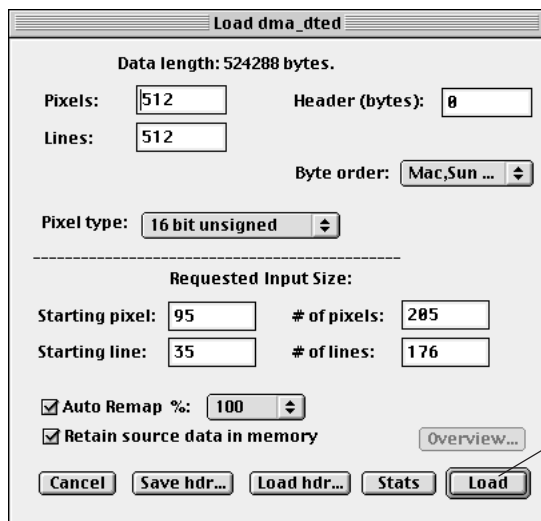

Overviews

The File->Open As menu items: Multiband, Raw and certain NITF and TIFF dialogs each contain an Overview button. Its purpose is to generate an image of reduced scaling that displays the entire or user specified subset of the image at a scale so that scrolling is not needed. A dialog is presented showing the nominal scale factor and the overview band which you may edit.

The Requested Input Size, the bottom portion of the dialog, specifies the area of the image used to generate the overview. The remapping options are used as normal to produce a decent appearing image. When generating a multiband overview, dialog item Starting band is the default image for the overview.

Once an overview image has been generated you can draw a rectangle (see section Selection Rectangle) on it that defines the portion of the source image to load by choosing menu File->Load Selection. Doing so brings up the dialog defining the image along with the Starting pixel, Starting line, # of pixels, and # of lines representing the image subset delineated by the selection rectangle. If no selection rectangle has been drawn on the over-view then the fields will be set to the spatial extent of the overview. For multiband images, the original Starting band and # of bands are not changed. You may edit any value before clicking the Load button.
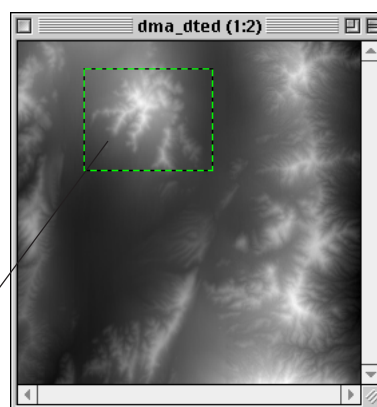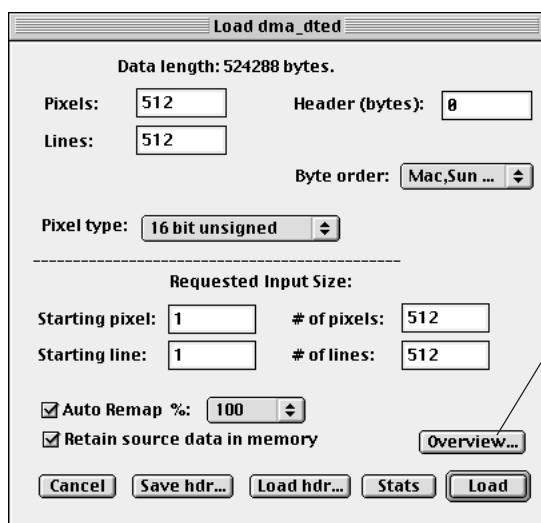

Choosing Overview *from the dialog at the upper left generated a 1:2 scale of the image shown above. A selection rectangle was drawn and* File->Load Selection *was used to display the dialog at the lower left. Note the new values that delineate the image shown below when the* Load *button was clicked.*

Saving Overviews

An overview image serves as a convenient method of automatically extracting the source file and saving it in a possibly different file format. This applies to both a selection rectangle and the entire image. Specifically, NITF, TIFF and Raw image overviews can be saved in any of those 3 file types. Cube image overviews can only be saved in cube image format. Note that the save is performed on the source image file depth regardless of the Retain source checkbox status when the overview was loaded. Any reference coordinates associated with the source image overview are correctly propogated to the saved image file. If the overview is color then the output is restricted to NITF and TIFF.

Shown below is a typical dialog that appears after selecting menu File->Save As when saving an overview image. In this case the the overview has a selection rectangle defined. Also, see sections: NITF Files, TIFF Files and Overviews for detail concerning the generation of overviews.
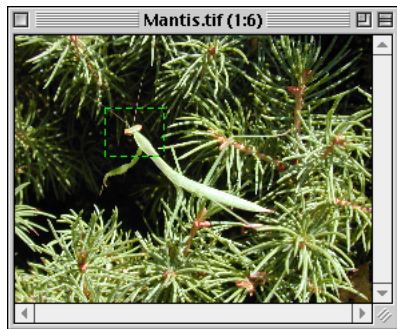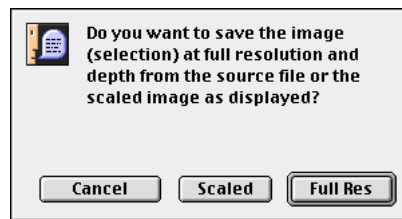
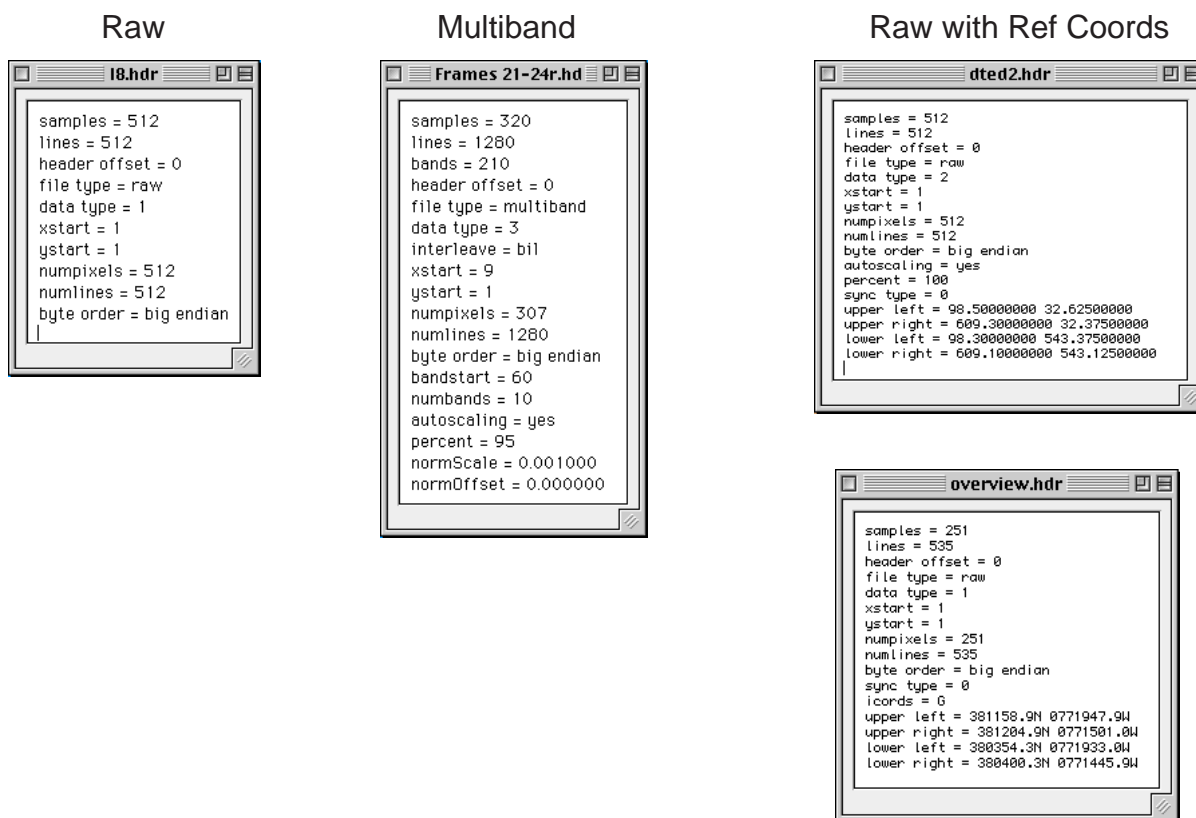*A true color 1:6 overview image with a selection rectangle.*
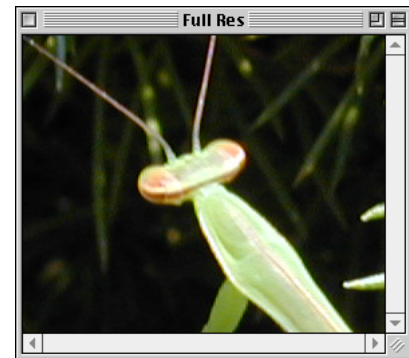
The reloaded image resulting from clicking the Full Res *button in the above dialog.*

Header Files

   As stated previously, each image file can have a unique ASCII text header file that contains all the image characteristics needed to load and display it. This file has the same name as the image but with a .hdr extension and must reside within the same folder as the image file. The two header files shown here (opened as Text files) correspond to raw and multiband examples respectively. They consist of key word assignments, one pair per line in no specific order (see section Keyword Format). A missing assignment defaults to "no" or zero. Any value can be overridden when brought up in the image format dialog and optionally re-saved. Although header files can be generated using HyperCube (see section Reading/Writing Text Files) and saved as "Text", it's easier to fill in the dialog fields and click the Save hdr... button.

   HyperCube can read certain ENVI® header files. Not every format is supported. The program first looks for the beginning ENVI® keyword and if found parses only the information necessary to conform to its own header parameters including wavelengths and bad band definitions. You are alerted if an ENVI® header file is being read. Saving (Save hdr...) information extracted from an ENVI® file writes a new HyperCube file <u>not</u> an ENVI® header file.

Raw	Multiband	Raw with Ref Coords

Raw — I8.hdr

```
samples = 512
lines = 512
header offset = 0
file type = raw
data type = 1
xstart = 1
ystart = 1
numpixels = 512
numlines = 512
byte order = big endian
```

Multiband — Frames 21-24r.hd

```
samples = 320
lines = 1280
bands = 210
header offset = 0
file type = multiband
data type = 3
interleave = bil
xstart = 9
ystart = 1
numpixels = 307
numlines = 1280
byte order = big endian
bandstart = 60
numbands = 10
autoscaling = yes
percent = 95
normScale = 0.001000
normOffset = 0.000000
```

Raw with Ref Coords — dted2.hdr

```
samples = 512
lines = 512
header offset = 0
file type = raw
data type = 2
xstart = 1
ystart = 1
numpixels = 512
numlines = 512
byte order = big endian
autoscaling = yes
percent = 100
sync type = 0
upper left = 98.50000000 32.62500000
upper right = 609.30000000 32.37500000
lower left = 98.30000000 543.37500000
lower right = 609.10000000 543.12500000
```

overview.hdr

```
samples = 251
lines = 535
header offset = 0
file type = raw
data type = 1
xstart = 1
ystart = 1
numpixels = 251
numlines = 535
byte order = big endian
sync type = 0
icords = G
upper left = 381158.9N 0771947.9W
upper right = 381204.9N 0771501.0W
lower left = 380354.3N 0771933.0W
lower right = 380400.3N 0771445.9W
```

Various examples of header files. The last 4 lines of the two right examples list the corner coords of the source image in an external reference system where each keyword specifies an X,Y pair (see next section, Keyword Format*). Any combination of 1,2,3 or all 4 keywords may be given. Selecting menu* Windows->Show Info *will display the computed reference coord as a function of the window cursor position. Section* Image to Reference Coordinates *describes how to edit these values.*

Keywords used in the header file are defined below. Each keyword is of the form:
        `keyword = value`. None of the words are case sensitive.

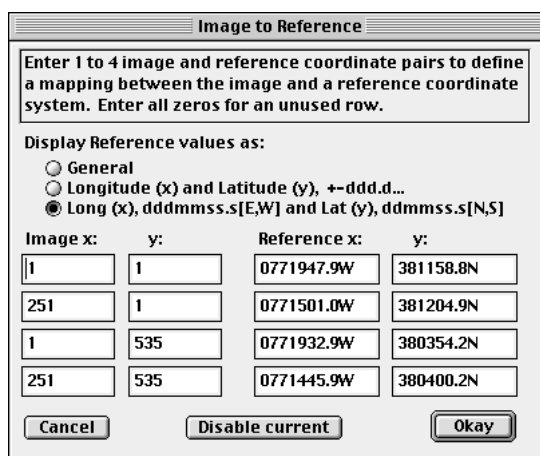| Keyword	Value
`samples`	number of source image pixels per line
`lines`	number of source image lines
`bands`	number of source image bands, if multiband or sync
`file type`	`multiband` `raw`
`data type`	`1` - unsigned byte `2` - unsigned 16 bit short `3` - signed 16 bit short `4` - 32 bit float `5` - 32 bit signed long
`interleave`	`bsq` - band sequential `bil` - line sequential `bip` - pixel sequential
`xstart`	starting pixel within the source image to load
`ystart`	starting line within the source image to load
`numpixels`	number of pixels to load beginning at `xstart`
`numlines`	number of lines to load beginning at `ystart`
`byteorder`	`little endian` - PC `big endian` - Mac, Sun
`bandstart`	starting band to load for display purposes
`numbands`	number of bands to display starting at `bandstart`
`autoscaling`	`yes` - remap image for display purposes
`percent`	% of remapping statistics to use for display purposes
`bandnormalize`	`yes` - remap each displayed band using its own statistics
`sigma`	number of standard deviations used by `bandnormalize`

`normScale`	scale factor to convert pixel values to [0,1] range
`normOffset`	offset to convert pixel values to [0,1] range
`sync type`	`tc` - true color synthetic cube `%` - classify overlay synthetic cube representing percentage `#` - classify overlay synthetic cube representing counts `d` - classify overlay synthetic cube representing thresholds
`retain data`	`yes` - set the "Retain source data in memory" checkbox in the various load dialogs.
`icords`	`G` - reference coords represent latitude and longitude formatted as ddmmss.sZ for latitude (y) where Z is N or S followed by dddmmss.sZ for longitude(x) where Z is E or W `D` - decimal degrees latitude (y) and longitude (x) Note the order: latitude value followed by the longitude value. Also, `icords` is not required to define a general reference.
`upper left`	x,y reference coord pair corresponding to pixel, line [1,1]
`upper right`	x,y pair corresponding to pixel, line [# image pixels,1]
`lower left`	x,y pair corresponding to pixel, line [1,# image lines]
`lower right`	x,y pair corresponding to pixel, line [# image pixels,# image lines]

Image to Reference Coordinates

You can specify the relation between image coordiantes and a reference coordinate system by including keywords such as `upper left`, `upper right`..., described in sections **Header Files** and **Keyword Format** or at any time an image window is active by choosing menu **Edit->Options->Image to Ref Coords...** . Some image formats such as NITF and TIFF also have reference coordinates in their headers. This brings up a dialog showing the current pairings of the image corner coordinates and the corresponding reference coordinates. If no relation has been defined then the mapping is the identity and the reference values will match the image values. Note that the dialog is completely general and you may specify other than the given corner coordinate pairings, e.g., a known image interior point and the matching reference point. From 1 (simple translation) to 4 (least squares affine) pairs can be specified. Set unwanted pairings to all zeros. Saving an image after defining an image to reference relation saves the equivalent <u>corner</u> <u>coordinate</u> to reference mapping in the image's header file. The option for formatted latitude and longitude is specific: ddmmss.sy where y is either N or S for latitude and dddmmss.sz where z is E or W for longitude. Some general reference values don't correspond to latitude and longitude and you will get a warning message when selecting from General to lat and long.

Note: Changing the image origin via **Edit->Options->[0,0] Image Origin** will disable all Image to Reference correspondences in effect and you will have to re-establish them.
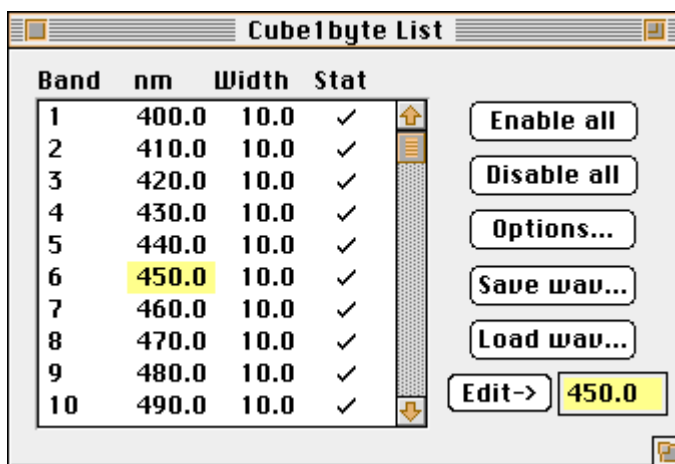

The correspondence between image coordinates and a reference system expressed in latitude and longitude. Clicking Okay *will define the relation which can be displayed dynamically in the* Info *window. This relation will be included in the header file if the image is saved. Clicking the* Cancel *button ignores any changes but does not remove an existing relation. Clicking the* Disable current *button removes the reference relation.*

Wavelength File

In order to perform any analytic operations on a multiband image, it is necessary to know the wavelength and band width of each of its bands. Similar to the *.hdr file there is an ASCII text file that specifies: absolute band number, **wave**length, band width and status for each band. It has the same name as the multiband image file but with a .wvl extension (previous versions of HyperCube used .wav extension which are still read correctly if the .wvl extension is not found). For an N band image there are N rows of values. Each row has 4 elements separated by a space or tab and followed by a carriage return. This can be generated using a spreadsheet or word processor and is completely editable in HyperCube. Below is the *.wvl file for the multiband example and the way it appears in the program for editing. In this case the wavelengths are uniformly spaced, the band width is a constant and every band is active. This is explained further in section Multiband Functions.

```
1    400    10  1
2    410    10  1
3    420    10  1
4    430    10  1
5    440    10  1
6    450    10  1
7    460    10  1
8    470    10  1
        •
        •
        •
70  1090   10  1
```
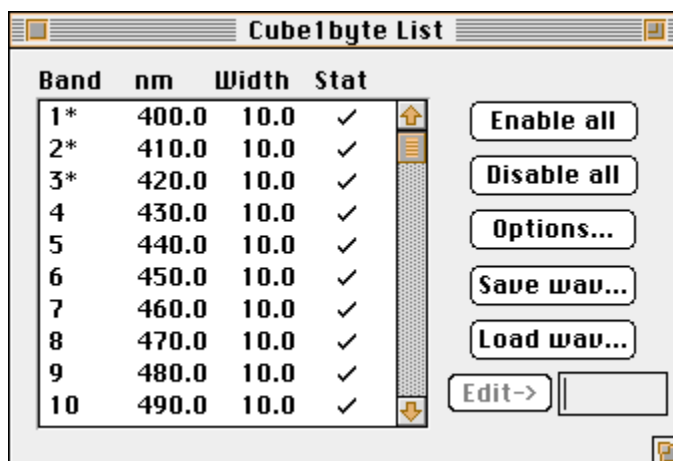
Band	nm	Width	Stat	
1	400.0	10.0	✓	Enable all
2	410.0	10.0	✓	Disable all
3	420.0	10.0	✓	
4	430.0	10.0	✓	Options...
5	440.0	10.0	✓	
6	450.0	10.0	✓	Save wav...
7	460.0	10.0	✓	
8	470.0	10.0	✓	Load wav...
9	480.0	10.0	✓	
10	490.0	10.0	✓	Edit-> 450.0

*ASCII *.wvl file*   *The HyperCube band list after selecting* Show Band List.

If the *.wvl (or the older *.wav) file doesn't exist for the multiband image you will get a warning message during the image loading process directing you to item Show Band List in the Windows menu where several options are available for creating/modifying one. If the band number is followed by an asterisk then that band wasn't loaded into memory. See section Band List under Multiband Functions for more detail.

The resultant band list after starting the multiband load at band number 4. Bands 1,2 and 3, which were not loaded, have been flagged with an asterisk.

Band	nm	Width	Stat	
1*	400.0	10.0	✓	Enable all
2*	410.0	10.0	✓	Disable all
3*	420.0	10.0	✓	
4	430.0	10.0	✓	Options...
5	440.0	10.0	✓	
6	450.0	10.0	✓	Save wav...
7	460.0	10.0	✓	
8	470.0	10.0	✓	Load wav...
9	480.0	10.0	✓	
10	490.0	10.0	✓	Edit->

Statistics

Shown below is the result of clicking the Stats button in the Image Format Dialog window for an 8 bit 512 by 512 raw image. Also see section Histograms.
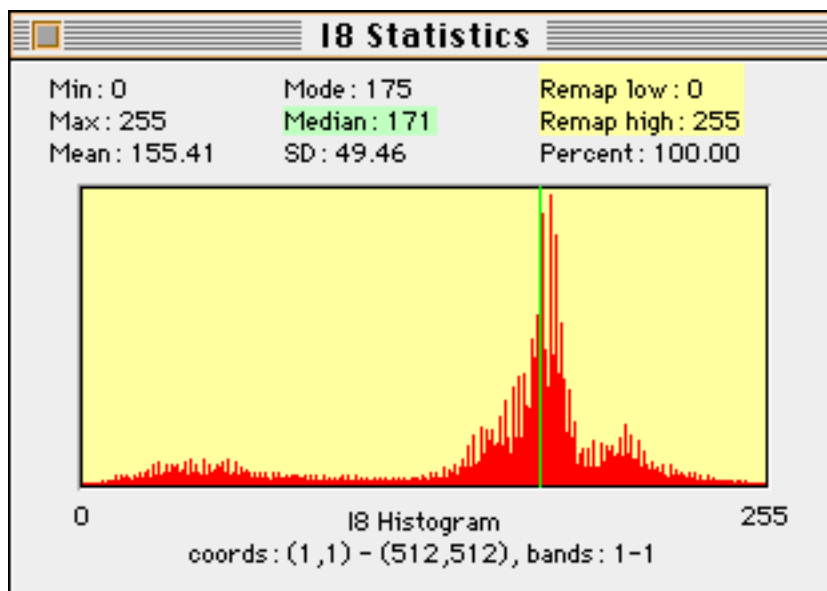
The histogram corresponding to an 8 bit image load .

The numbers shown for Min, Max, and so on are the values for the pixels in the image file within the rectangular area chosen by the user for subsequent loading. In this case they are confined to 8 bit unsigned [0,255] pixels. The remapping limits extend over the full range of the input data because the percent default is at 100. Another more complex example appears on the following page.

The statistics window is important for visualization because based upon its values the image file pixels are remapped to the 8 bit [0,255] display range. Explicitly, the image file interval:

[pixel_median - %/2, pixel_median + %/2] is mapped to [0,255].

Values < pixel_median - %/2 are set to zero.

Values > pixel_median + %/2 are set to 255.

If the resulting loaded image appears too bland (washed out) then the auto remapping percentage should be made lower using the % popup. If the image appears too contrasty then the percentage should be made larger.
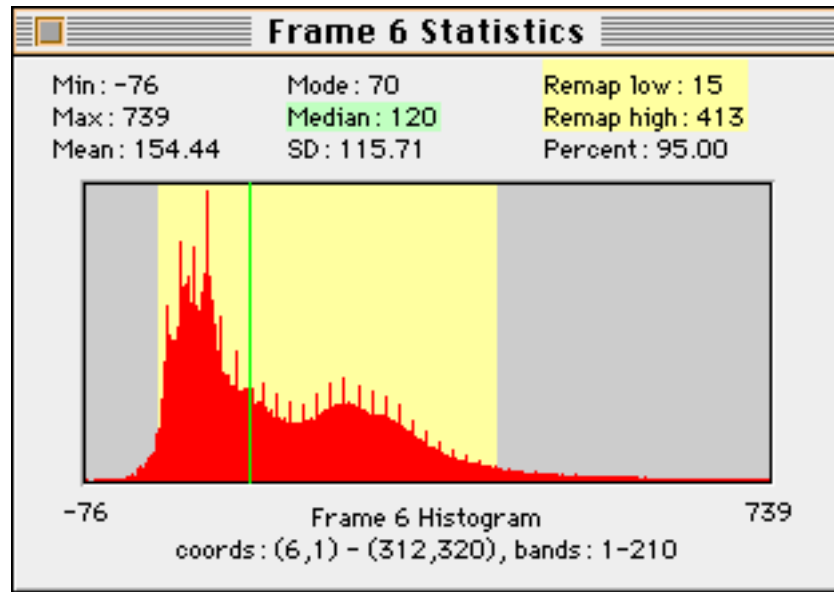
The resultant statistics window corresponding to the previous multiband example using all 210 bands with the 95% remapping interval highlighted.

Below are snippets of the example multiband data after being loaded with 3 different auto remapping percentages.
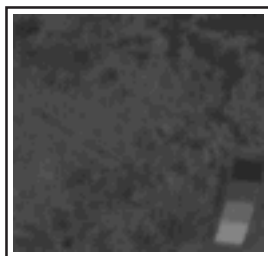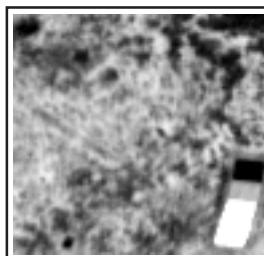
100% Interval   *95% Interval*   *70% Interval*

Regardless of how the imagery is remapped for display purposes, operations such as plotting and spectral classification can operate upon the image file as well as the converted display values. So although saturation and underflow typically occurs when displaying multiband images all of the arithmetic can be performed on the full range of the file data.

Also, if Remap Bands Individually is chosen in the load cube dialog then each band will be correctly gray mapped but at the expense of maintaining their relative brightnesses. Again, this will not affect any classifications or plotting. If Detailed Messages is checked on in the Options menu two scrollable lists will be output showing the individual statistics and remapping coefficients for each band.

Image Operations

Zoom

Image zooming is selected via menu Image->Zoom. It is automatically enabled whenever the active window is an image window, otherwise it is disabled. Each Zoom In magnifies an image by 2 until it reaches 16 times then it is disabled. Each Zoom Out de-magnifies by 2 until the magnification is unity. The current zoom factor is displayed in each window's title bar. Zooming is performed by pixel replication and is a "virtual" image in that a saved zoomed image is un-zoomed. To generate a "real" magnified image use Magnify in the Image menu. Printing or copying a zoomed window does produce a zoomed output. The keyboard equivalents for Zoom In and Zoom Out are command + and - respectively.

Selection Rectangle

Many image operations can be constrained to a subset of an image by drawing a selection rectangle within the image window. This is done by positioning the cursor to one corner of the area, pressing the mouse button and dragging to the opposite corner and releasing the mouse button. When completed the area will be delineated by a rotating marquee at the boundary as shown here. A single mouse click

© 1935 Paramont

(double click in the Windows version) anywhere within the image window or pressing the delete key or choosing Clear from the Edit menu will deselect the rectangle. The rectangle may be repositioned by pressing and holding down the mouse while the cursor is within the selection boundary and dragging to a new position and then releasing the mouse. The cursor changes to a multiple arrow while dragging. Each image window can have at most one selection rectangle. It's possible to unintentionally hide a selection if a window is re-sized. Scrolling a window will bring the selection area back into view.

Copy, Paste & Duplicate

HyperCube allows copying and pasting entire images or selection rectangles between the Mac clipboard and its own image windows as well as other application's windows. Only Pict type data is recognized. This is a standard clipboard data type used by every application. In order to perform a copy draw a selection rectangle about the image area to be copied, then choose Copy from the Edit menu or the keyboard equivalent, command C. If there is no selection rectangle active or the window contains a plot or list or some other non-image then the entire window is copied to the clipboard. To paste something already in the clipboard, click anywhere within the destination window to activate it and choose Paste from the Edit menu or its keyboard equivalent, command V. Paste will not be enabled unless there is Pict data or color pixel value in the clipboard. The pasted data appears in the center of the viewable area of the front most image window with a selection rectangle around it. It does not paste into an existing selection and if there was one it just disappears. If the target image is too small to hold the paste then the clipboard data is re-sized to fit within the image boundaries. The pasted data can be moved by clicking and holding down the mouse within the selection rectangle and dragging it to a new position. The cursor changes to a hand during

the repositioning. A single click anywhere within the image window embeds the paste. Pressing the delete key or choosing menu Edit->Clear erases the selection without pasting it.

   If File->New... is chosen and there is something in the clipboard to paste then the default size of the new window will be set equal to the size of the image in the clipboard. Performing a paste will copy the clipboard to the new window.

     A copy of the currently active image window or selection rectangle can be made by choosing the Edit->Duplicate menu. This option is available for almost every type of window. The duplicated image will have the same name as the original followed by the word "Copy". Some HyperCube operations, such as constructing a cube, automatically make image duplicates. Also, see section Saving and Copying Scrollable List.

     Any time that the cusor is over an image window, regardless if the window is front most, the current pixel value may be copied to the clipboard by also pressing the option key, e.g. command+option+C combination (Ctrl+Alt+C in <u>Windows</u> version). This value may subsequently be used as the selected color in many HyperCube functions, (see sections Image to Mask, Adjacency, Changing Selection Point Colors and Color to Name Editor).

Gray Mapping

     The contrast and brightness of the active gray scale or color image window may be interactively changed by selecting menu Image->Gray Map. This opens a window containing controls to modify the pixels in the image and a graphical plot of the current transforma-
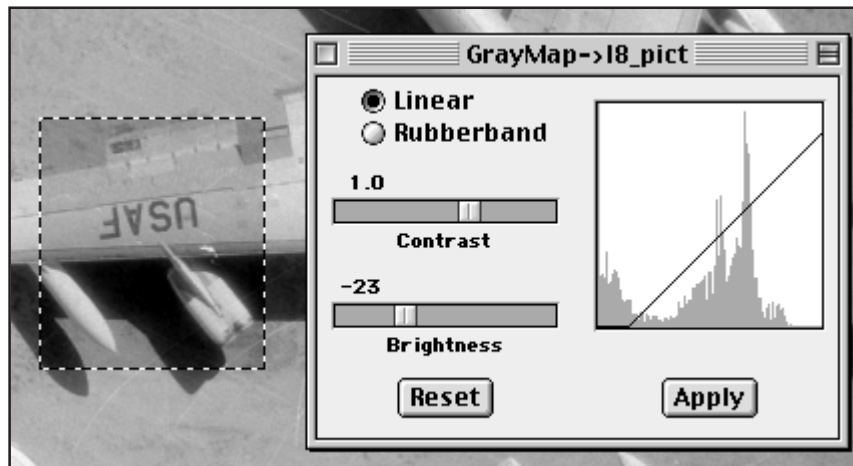
The effect of modifying the brightness within a selection rectangle.

tion and resulting histogram. The GrayMap window applies only to the image window that was active (front most) when Gray Map... was selected. However, every image window can have its own gray mapping operator. For Linear mapping the Contrast and Brightness sliders control the slope and y intercept of the graphical plot respectively. The horizontal plot axis represents the input pixels [0,255] and the vertical axis represents the remapped pixel values, also in the [0,255] range. Initially, the slope is 1 and the intercept is 0 signifying that the input is identically mapped to the output. The contrast range is [-9,9] and the brightness range is [-56,56]. Negative slopes produce complementary pixels. Above is an example of gray mapping applied to just a selection rectangle within an image. If there is no selection

rectangle then the entire image is gray mapped. Clicking the Apply button embeds the gray mapping into the selection area, or the entire image if no selection. There is one exception: an 8 bit color indexed image is remapped entirely regardless of any selection rectangle. If a GrayMap window becomes hidden behind another window you can bring it to the front by selecting its name from the Windows menu. As long as the Apply button has not been clicked the identity gray mapping can be recovered by clicking the Reset button. Show Histogram dynamically displays the entire image's histogram as a function of the current transformation starting with the true historgram of the image.

Rubberband gray mapping doesn't use the slider controls. It allows you to position the cursor within the plot area while holding down the mouse button. The graph then tracks the cursor and consists of two slopes with the cursor at their break point. Rubberband mapping is very useful in bringing out shadow detail without completely washing-out the surrounding area. An example of this is shown below.
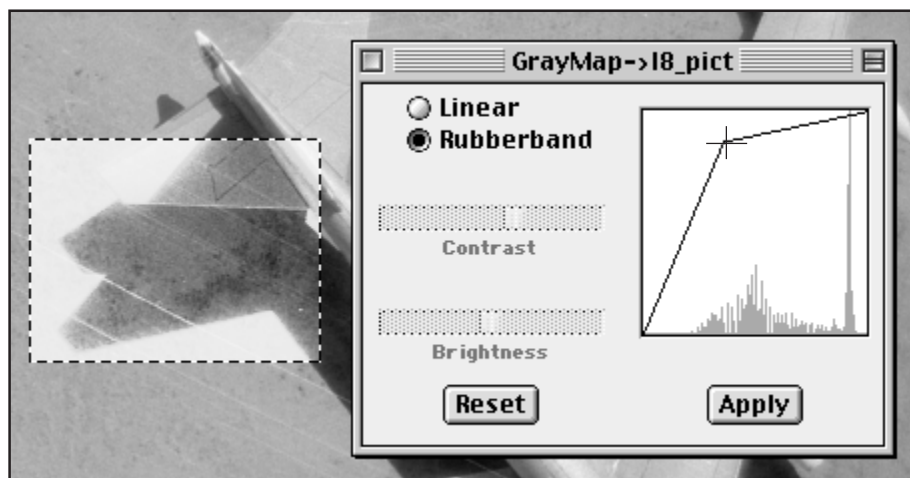
Rubberband gray mapping being applied to a selection area. The sliders are not used and are grayed out.

Color Mapping

You can modify the color balance of an image by selecting menu Image->Color Map. It is similar to gray level mapping (Gray Map) except that it only operates on color images. Sliders are used to independently modify the red, green and blue brightness and contrast of an image while showing the effect on the 3 historgrams. Note that subtracting a color is the same as adding its complement and vice versa: adding is subtracting the complement. The controls and buttons perform the same as gray mapping with the same exception: Apply modifies an entire 8 bit indexed color image regardless of any selection rectangle.

An image may have only one operator attached to it at a time. Specifically, Gray Map, Color Map, and the two filtering operators: Static and Dynamic (described in later sections) are all mutually exclusive.
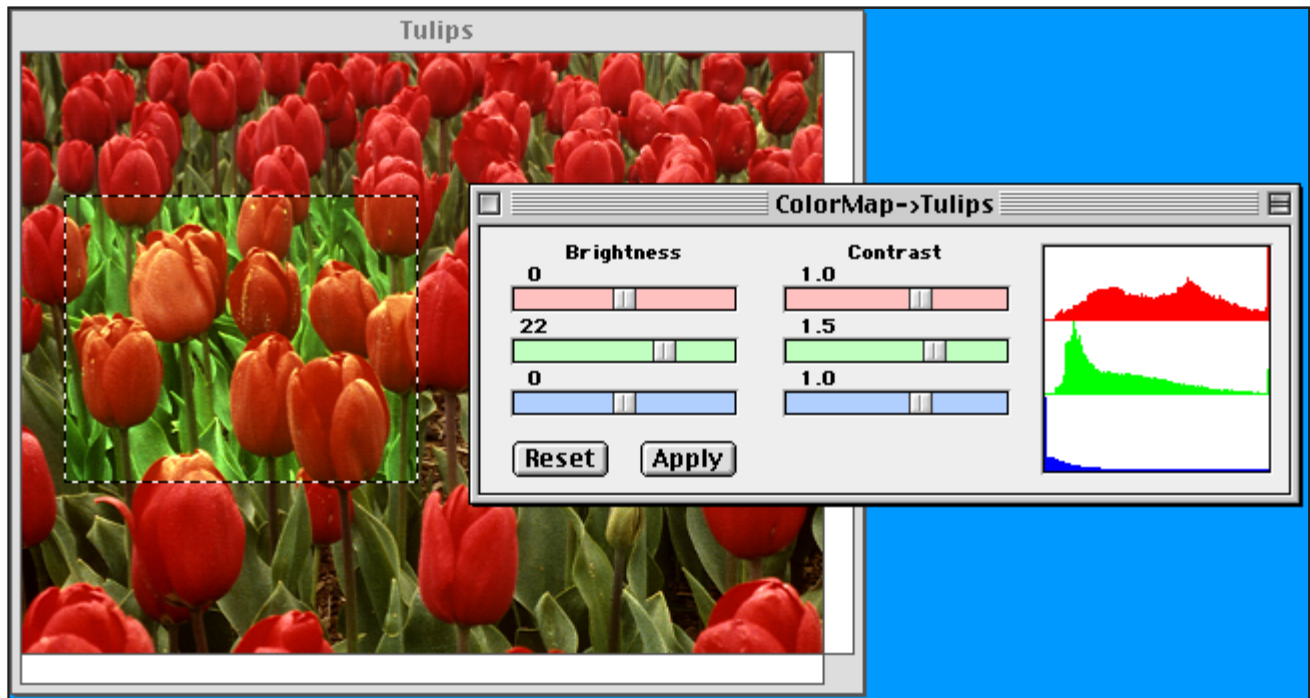
Color balance emphasizing green within a selection rectangle.

Flips & Rotations

Each loaded image can be flipped and rotated in orientation. As shown below there are 3 kinds of flips and 3 rotations. The Left->Right flip changes the order of the pixels within a line and the Top->Bottom flip changes the order of the lines. Band Order is only enabled for multiband images and re-sequences loaded bands but not their wavelengths. Therefore, the last band becomes the first band and the first band becomes the last band. Flips and rotations are global operations and ignore any selection rectangles.
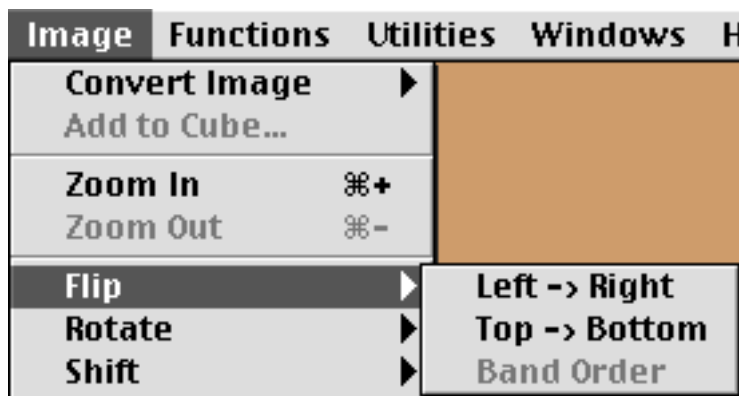
Flip, Rotate *and* Shift *appear in the* Image *menu.*

The Rotate menu item is similar to Flip. Rotations may be 90, 180 or an arbitrary number of degrees. 90 and 180 operate on an entire image (including multiband) while General can rotate a selection rectangle as well as the whole image and uses the currently selected interpolation method (see section Magnify). Note that rotating a non-square image results in interchanging the number of pixels and lines. Also, flips and rotations disassociate the new image geometry from the original image file geometry. Therefore, library matches and classifications of multiband images will be erroneous as well as spectral plots of their files.

The Reformat Cube File function in the Utilities menu is used to change the internal structure of the multiband image <u>file</u> itself not just what is loaded in memory. It also regenerates the *.hdr and *.wav files as necessary.

Shifts

The Image->Shift menu item operates on whole images and selection rectangles. All of the shifts are cyclic so that pixels shifted off one end appear on the opposite end. If a selection rectangle is active then all shifting is confined to the rectangle. Since the shifts are invertible they are applied to the image immediately and can be removed by applying the opposite shift. Left and Right shifts can be applied to All lines or just the Even lines. They are useful in correcting interleaving errors in captured video images. Below is an example of such an image before and after shifting only the even lines. Shifts applied to a cube only affect the currently displayed band.
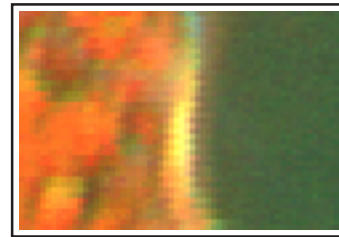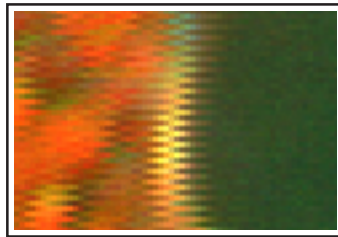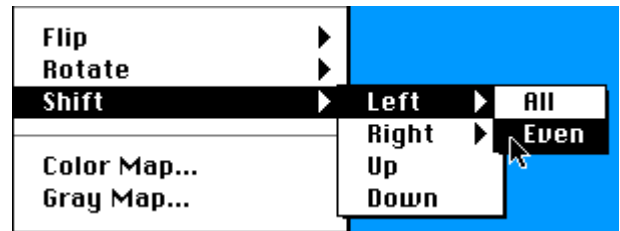
Magnified captured video frame before and after Shift Right Even lines.

Filtering (Spatial Domain)

HyperCube provides several methods for filtering an image in both the spatial (pixel) domain and the frequency (Fourier) domain. The later is described in the Image Functions section. Spatial filtering convolves the image pixels with various user selected matrices and HyperCube provides two methods: a 3x3 interactive kernel and a 5x5 static kernel. The 3x3 contains sliders for altering the matrix elements while the 5x5 uses edit fields and operates as a one-shot. You may save the 5x5 coefficients and read existing 5x5 convolver values including those saved by Photoshop™.

Each type of convolver is an operator to the specific image window that was active when Image->Filter was selected from the Image menu. Closing the image also closes the filter. If a selection rectangle is present on an image then the filter is constrained to that area. The filtering does not become permanent until the Apply button is clicked.

Filtering is an image operator and as stated earlier an image may have at most one operator at a time, e.g. if Gray Map is active on an image then it must be closed before any filtering can be applied.

The various sliders on the 3x3 convolver have the following actions:

Diagonal	Modify the 4 corners, (1,1), (1,3), (3,1), and (3,3)
Edge	Modify only the (1,2), (2,1), (2,3) and (3,2) elements
Center	Modify only the (2,2) element

Note that since the sliders produce positive and negative values many positions don't produce meaningful image manipulations. The Sharp <--> Blur slider tries to solve this problem by parametrically modifying the other 3 sliders to produce a continuous range from sharpen to blur. The 3 by 3 matrix window shows the instantaneous convolver coefficients as depicted in the figure below.
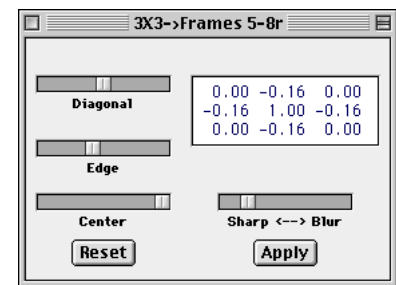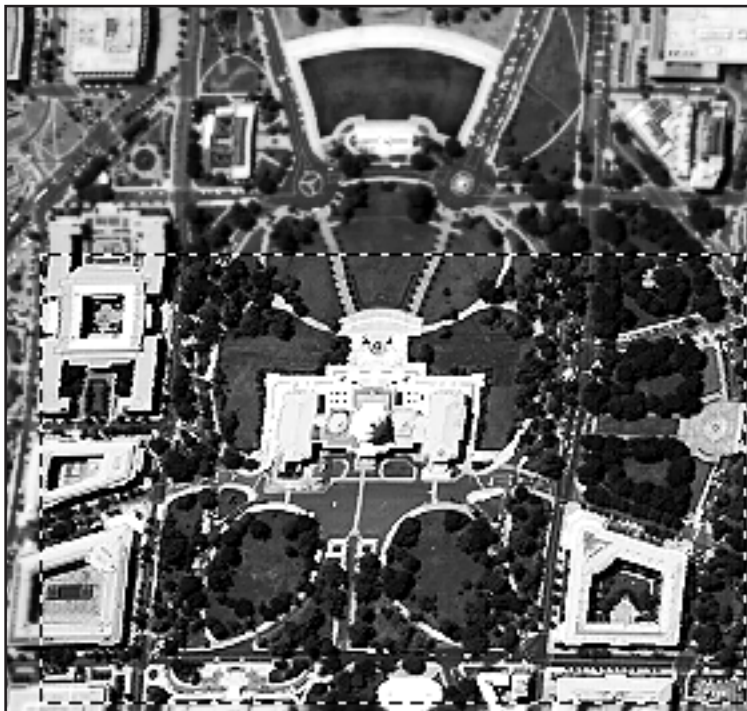
The 3x3 convolver applying a sharpening function to a selection rectangle within an image.

The 5x5 static convolver, shown at right, allows you to input 25 separate coefficients. All of the coefficients must be integers. The Scale value is divided into the coefficients and the Offset value is added to the final resultant pixel. The popup menu has several predefined kernels which may be modified. A loaded or save kernel will occupy the User menu slot. Since there are so many degrees of freedom it's easy to generate meaningless pixels and experimentation helps. Any set of coefficients may be saved via the Save... button and retrieved with the Load... button. The coefficients are compatible with those produced by Photoshop™ and HyperCube will recognize the later.
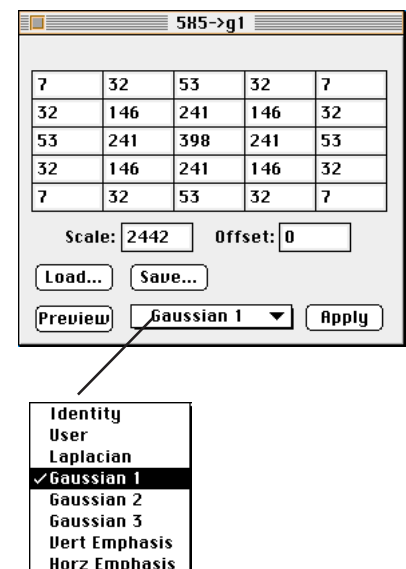
Magnify

A whole image or the contents of an image selection rectangle, including the current cube face, can be magnified producing a new independent image. The magnification range is mostly unlimited as long as the resultant image is at least 32 by 32 in size and there exists enough RAM. The type of interpolation can be replication, bilinear or cubic resampling. Menu item Image->Magnify toggles between Image and Selection depending on whether or not a selection rectangle is exists on the object image window (see section HyperCube Menus). A submenu specifies several magnification factors and permits you to enter an arbitrary factor that is preset to .75x. The type of interpolation is a nested submenu, Edit->Options->Mag Interpolation, and once selected remains so until changed. The initial interpolation is set to Replicate. You must remember that magnifying 8 times generates 64 times as many pixels as the source. Included below are examples of a 4x magnification using each type of interpolation. Replication is best for viewing the actual pixels while bilinear and cubic produce photographic like enlargements. Also see the HyperCube section describing Warp.
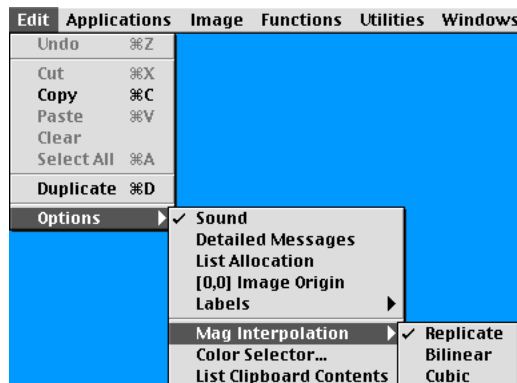
A 4x magnification applied to an image subset utilizing all three interpolators. The original is shown at 2x for clarity.

Replicate
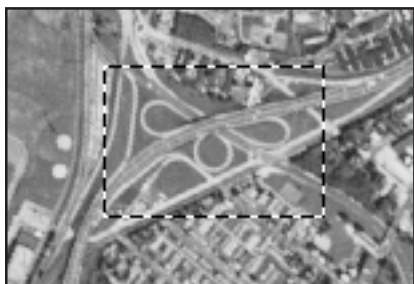
Original selection

Bilinear

Cubic

Info Window

HyperCube contains one window, Info, that is always available and dynamically displays information about any of the other HyperCube windows. The kinds of information depend upon which window the cursor is in. When the program starts, Info is hidden and can be shown at anytime by selecting menu Windows->Show Info. It is not a floating window when running in the Classic Mac OS and if it becomes obscured by another window can be brought to the front by Show Info. It is a floating window in Mac OS X and in Windows. To hide it click its close box. To print it, use the keyboard equivalent command P instead of the mouse so that the cursor doesn't move off its current position and change the Info values.

The types of information displayed include the amount of remaining free memory, the type of window, and the position and current gray or color values of the pixel beneath the cursor. The origin of all windows is [1,1] at the upper left unless [0,0] Image Origin is selected from the Options menu. When the cursor is placed upon an image cube Info shows a mini-plot of a vector through the loaded cube bands at that position. And when placed on a histogram plot it shows the bin number. Various Info window examples appear below.
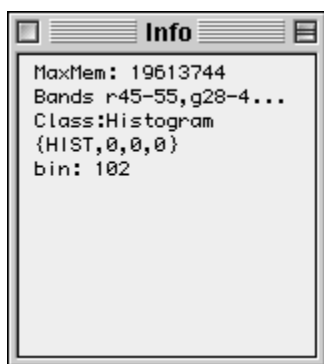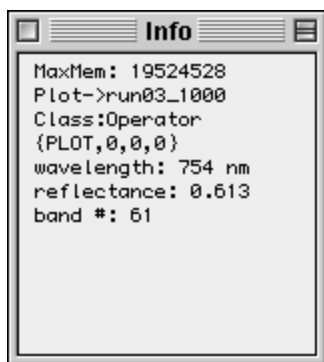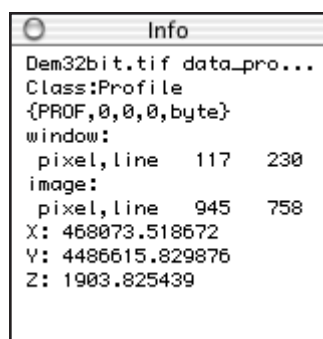

Cusor positioned over a histogram plot.
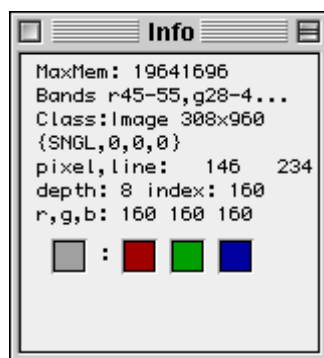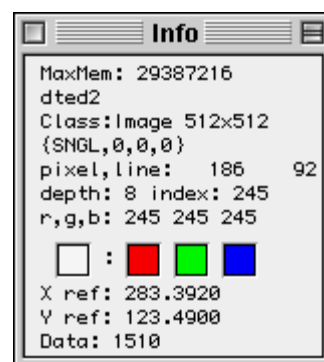

Cursor over a gray scale image.
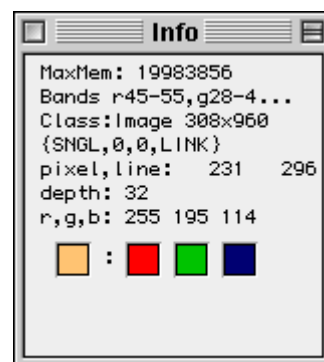
A true color image pixel with the red, green and blue color components.

Cursor over a spectral plot window.

Cursor over a profile plot showing the corresponding x,y,z image values for a point on the profile line.


Over a Raw *image with retained data and external reference coords (see section* Header Files*).*

Image Conversions

Components to Color

Three gray scale images are combined to form a composite true color image and is chosen by selecting Image->Convert Image->R,G,B to Color or H,S,I to Color (Hue, Saturation and Intensity to color). Since any three gray scale images can be used, including bands of a multiband cube image, HyperCube displays a dialog window asking you to specify which image represents which of the three colors. The three components do not have to be the same size, the true color result will be their minimum size. If all three images are bands from the same image cube the Link to Cube check box is enabled and if selected the resultant color image will be logically connected to the parent cube image. This image can then be used to select points for subsequent spectral plotting and classification (see section Classify). If red, green and blue are all cube bands (not necessarily from the same cube) then Entire Cube may be selected and a synthetic color cube will be automatically generated in which each band consists of the corresponding components (bands) from the cubes. The number of bands in this cube will be the minimum of the number of bands currently loaded from the source cubes.
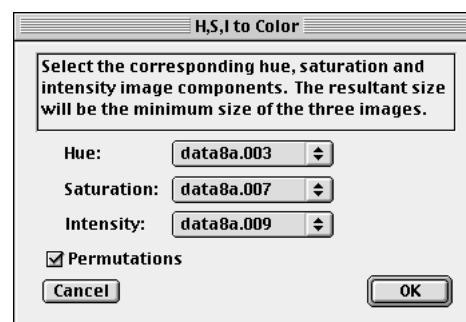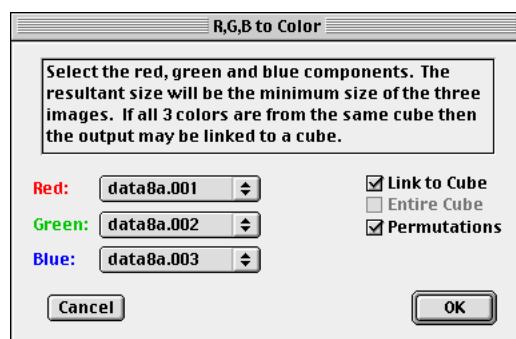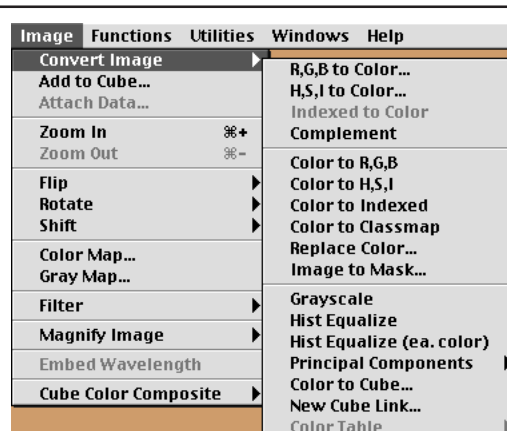
The left dialog shows the selection of three bands to be composited to form a true color image. The right dialog similarly forms a true color image treating the components as if they represented hue, saturation and intensity (also see Applications->Stereo Mates*). If* Permutations *is checked then all 6 combinations of the source planes will produce 6 color images (saves having to specify each combination).*

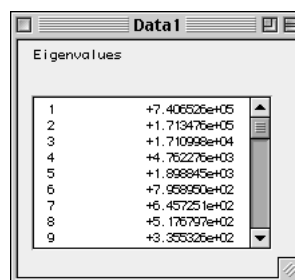## Image to Various Components

These are inverses of the previous functions. Any true color image window, i.e. those that show a depth of 24 or 32 in the Info window, can be separated into 3 same size gray scale images representing the red, green and blue or hue, saturation and intensity, components respectively. The original image is not affected. To perform the conversion activate the color image window and choose Image->Convert Image->Color to R,G,B or Color to H,S,I . The 3 new images that are formed are not connected in any way to the original color image or to themselves. You can make them into a multiband image by the Plane to Cube menu function described in a later section.
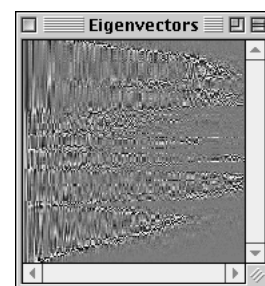
Image to Principal Components

Image->Convert Image->Principal Components uses the Hotelling transform to convert a color or cube image into its principal components. Either the covariance or correlation matrix statistics can be used as the basis for the transform (and optionally saved). If the source is a cube then you will be presented with the following dialog allowing you to transform either just the bands that are currently loaded into memory or the actual cube file. The result of transforming an image cube <u>file</u> is a cube file that can be treated like any image cube to perform classifications and "spectral" looking plots.
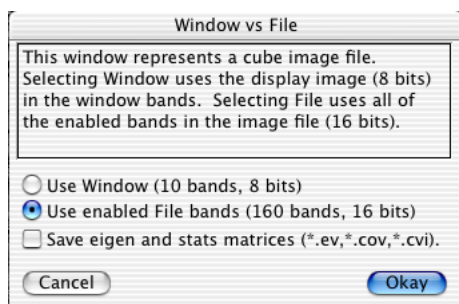
Selecting Window *transforms only those bands presently in memory using just 8 bit data. Selecting* File *transforms the image cube file and produces a new cube file. The bands are the principal components and are ordered in decreasing eigenvalues. The* Save... *check box writes the various matrices to an ASCII file. See section* Apply Transformation *for format details.*
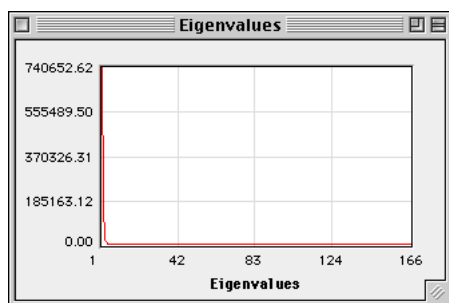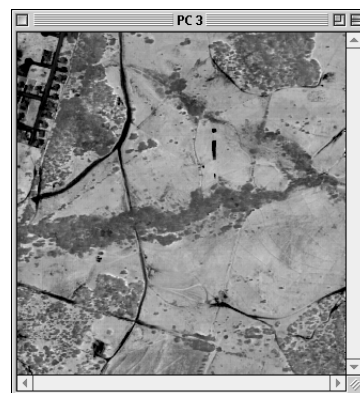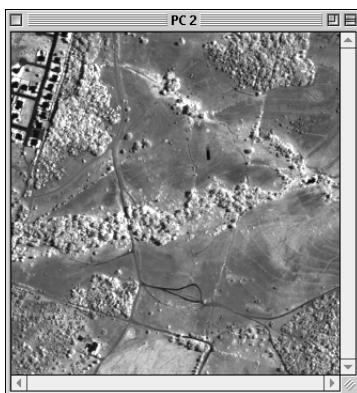
Shown at the top are the first 3 principal components generated from 166 enabled bands of a 210 band image. The plot and listing of the eigenvalues are also output as well as an image representation of the eigenvectors. The later may be examined via a line profile plot, histogram or using the cursor with the Info *window (see section* Apply Transformation *for format details).*

Indexed to Color

An indexed, or pseudo, image is an 8 bit image similar to gray scale but one in which the pixel value is an index into a color table. Therefore it has at most 256 distinct colors. A gray scale image is a special case of an indexed image in that its color table has the same red, green and blue intensity for each of its indices and is the actual pixel value (HyperCube checks all indexed images to see if they are actually gray because this greatly simplifies a lot of arithmetic). If the currently active window contains an indexed image then you can select Image->Convert Image->Indexed to Color. This will generate a new true color image. If the original image was gray then the new image will also be gray with equal red, green and blue components at each pixel. Note that on the Macintosh a full true color image is 32 bits in depth, not 24 as might be expected. The extra byte is for special effects. HyperCube also handles the Macintosh 16 bit (thousands) color image: three 5 bit color components plus a special effect bit.

Special note, Mac OS X only: Eight bit pseudo color Pict images are converted to true color when loaded. This does not apply to Pict 8 bit gray images or any other file types.

Color to Indexed

This is the most difficult type of color image conversion because there is no one to one transformation. A typical 512 by 512 true color image could have over a quarter million colors which must be reduced to only 256 colors. HyperCube tries to find the best 256 but sometimes the results are not acceptable. Normally, true color to index should only be used when the source image has very few distinct colors or to save storage as in making a true color image fit on a floppy disk. Its necessity has become less valid with JPEG compression (see section JPEG Files). You can select the Image->Convert Image->Color to Indexed menu when the front most window contains a true color image.

Gray Scale & Changing the Color Table

You can convert the active image window to a new 8 bit gray scale image by selecting the Image->Convert Image->Grayscale submenu. If the image was already a gray scale then a copy is formed. If the image was color, either indexed or true, the three color components are combined together to form a single intensity by the following formula used to convert NTSC color signals to black and white:

$$gray = .30*red + .59*green + .11*blue$$

This method yields a more natural gray image then using equal weights of the 3 components.

HyperCube contains a submenu called Color Table under Convert Image showing several color mappings that may be applied to an 8 bit indexed image. The top one overwrites the image's color table with a linear gray ramp forcing an indexed color image into some gray appearing image. The second strip complements the existing table so that selecting it twice restores the original table. The remaining 3 menu options overwrite the existing color table and color code the image pixels so that pixel (index) zero represents the left most color and pixel 255 the right most color. Also see the Classify section describing output products.
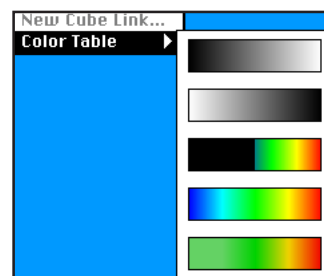

Any specific color or gray value within an image may be substituted with another value. The result is a same sized image with only that modification. If the image is a cube then a copy of the band face is used as the source. Choosing Image->Convert Image->Replace Color... brings up the following dialog with the last two colors or gray values, if any, that have been first selected by the command+option+C key combination or Ctrl+Alt+C in Windows version (also see section Copy, Paste & Duplicate). The first selected value will be Color 1, the color to search for, and the second as Color 2, the replacement color.

If Replace all gray... is checked then Color 1 is ignored and every gray value (non-color) is replaced with Color 2. This is useful for converting a color overlay to a mask.
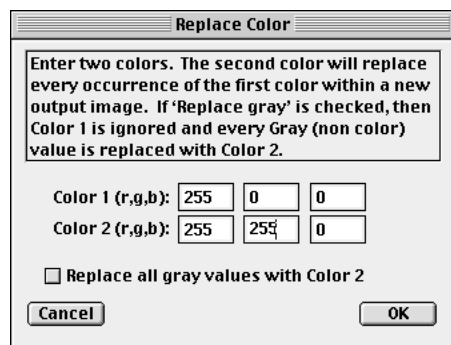
The Replace Color *dialog was used twice to convert the image at below left, a classification mask, to the image at its right. The dialog shows the first conversion, red to yellow. A second conversion using the output from the first replaced the black background with gray [128,128,128]. The intermediate yellow and black image is not shown.* Replace all gray... *can be used to convert a color overlay to a color mask. See section* Image Products.
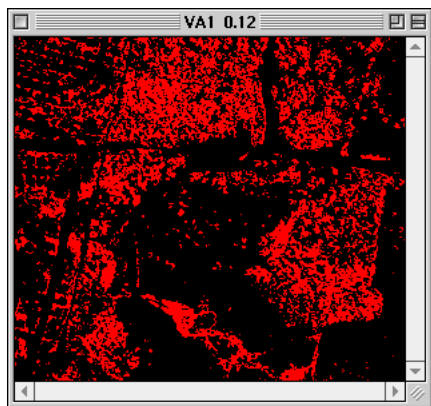
Original spectral classification mask as output from Classify.
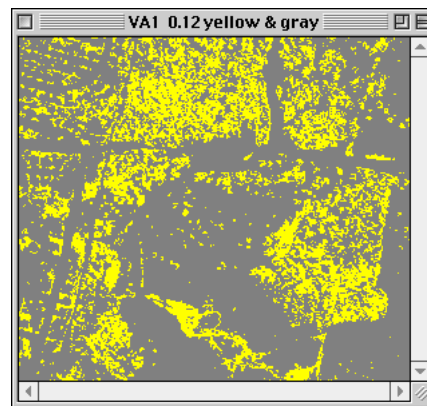
*After two successive uses of* Replace Color.

Image to Mask

An image can be converted to another image that corresponds to a specific gray level or color within the image. Two types of output are possible. The first (Binary) produces a same size image (mask) that consists of white (255) everywhere that specific color (gray level) appeared in the source image and black (0) everywhere else. The second type (Color separation) consists of the chosen color in place of white and black everywhere else, i.e. only the color selection itself remains. This is most useful where the source image does not contain numerous colors (gray levels) such as when separating a specific classification color from a color composite produced with the Classify function. The initial color (gray level) to be extracted is automatically preset to the last comnand-option C key combination used to copy a pixel color to the clipboard (see section Copy, Paste & Duplicate).

When an image is frontmost, choose Image->Convert Image->Image to Mask. This will bring up the following dialog window:


The respective red, green and blue components of the color to be extracted in generating the mask images in the examples below. Mask Type Binary *produces white wherever the rgb pixel value is [241,167,185] while* Color Separation *outputs the actual pixel value. If the image to be converted is a gray level image then the dialog will instead solicit a single gray level with the same options.*
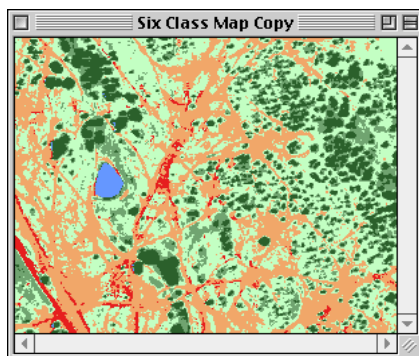
A source true color image that was derived from a spectral classification. The cursor was positioned over a road and the command-option-C *key combination pressed to store the pixel color into the clipboard <u>prior</u> to choosing* Image->Convert->Image to Mask... *from the menu.*
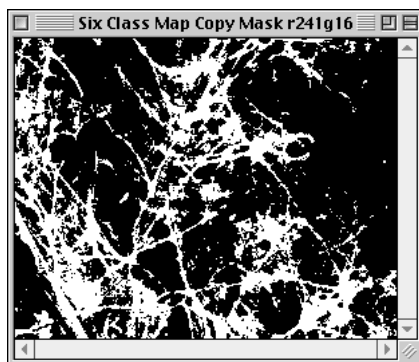
Binary Mask Type *consisting only of black (0) and white (255) values.*
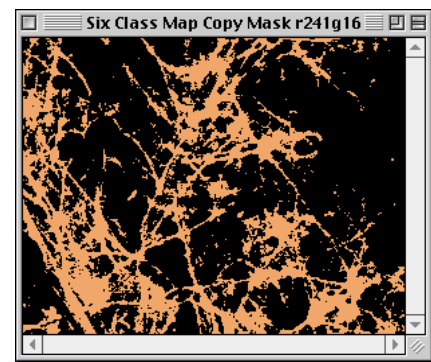
Color separation Mask Type *consisting of black and the extracted color.*

Histogram Equalization

HyperCube has two functions within the Convert Image submenu to perform a histogram equalization of an image. In each case a new image is generated leaving the source image intact. Any gray or color image may be the source. If the source is a pseudo color image then the equalized output is converted to true color, otherwise the output image is the same type and pixel depth as the source. Both functions attempt to construct an image whose histogram resembles a uniform distribution. Generally, this is not possible. For example, a non-uniform 256 bin histogram (8 bit image or a color component) can't be mapped to a uniform 256 bin histogram, there will be "holes". The method used by the two functions differs as follows: Hist Equalize attempts to keep the hue (color) of the image the same allowing the contrast and brightness to change while Hist Equalize (ea. color) performs the equalization independently on each color (red, green, blue) and then combines the results. The former method appears better but each should be tried. It is also possible to combine, e.g. averaging, the source and output to produce a more desirable effect (see section Image Arithmetic). If the source image has a selection rectangle present then the equalization is based only on the selection rectangle statistics but is applied to the entire output image.

Shown below is an original source image along with the output images generated by the two methods.
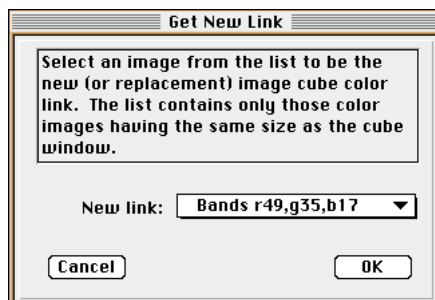

Original true color image.
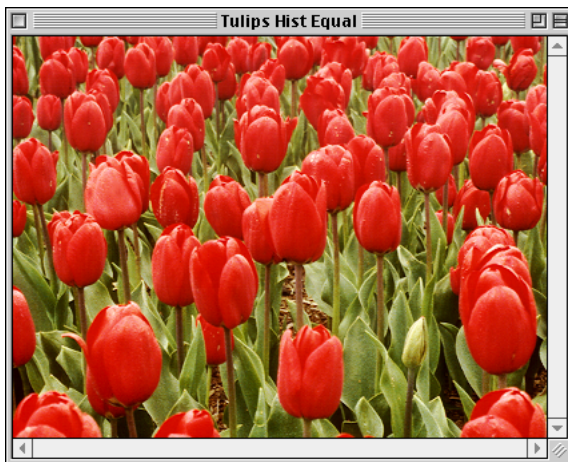
The left image is the result of using Hist Equalize *which attempts to preserve the color balance. The right image was formed via* Hist Equalize (ea. color) *which doesn't.*

Changing the Cube Color Linked Image

The current, if any, linked color composite image can be switched to another existing color image as long as it is the same spatial size by selecting Image->Convert Image->New Cube Link. The following dialog then appears listing candidate images. Any previous link will be disconnected and any current selection points will be transfered to the new link.

```
╔══════════ Get New Link ══════════╗
║ Select an image from the list to be the  ║
║ new (or replacement) image cube color     ║
║ link.  The list contains only those color ║
║ images having the same size as the cube    ║
║ window.                                    ║
║                                            ║
║   New link:   [ Bands r49,g35,b17    ▼ ]  ║
║                                            ║
║   [ Cancel ]              [   OK   ]       ║
╚════════════════════════════════════╝
```

Color to Classmap

A true color image can be transformed into a classmap in which the first 1024 colors (gray shades are excluded) are assigned classes 1 thru 1024 by selecting menu Image->Convert Image->Color to Classmap. The classmap can then be modified using all of the features in menu Utilities->Classmap Editor (see section Classmap Editor) If the source image is pseudo color you will be prompted to convert it to a true color image. In this case, there will be at most 256 classes. Since a typical color image usually contains more than 1024 colors this conversion is most useful after masking and/or image arithmetic functions to reduce the number of colors.

Attach Data

Sometimes it's convenient to merge 2 disparate types of images together. For example: a classmap (color coded features) and a matching elevation image (data values). This is accomplished by selecting the data image window to make it active and then choosing menu Image->Attach Data... . This brings up a dialog listing all of the candidate images. An image is a candidate to receive data only if the number of lines and pixels is the same as the data image. An image is a data image only if it was loaded with the Retain source... option selected (see sections: Raw, Multiband... and TIFF Files). After data has been attached to an image the Info window will show the image color/gray values and the corresponding data value as the cursor is moved over the image (see section Info Window).

```
Image   Functions   Utilities
  Convert Image          ▶
  Add to Cube...
  Attach Data...
  Zoom In            ⌘+
  Zoom Out           ⌘-
  Flip                   ▶
  Rotate                 ▶
  Shift                  ▶
  Color Map...
  Gray Map...
  Filter                 ▶
  Magnify Image          ▶
  Embed Wavelength
  Cube Color Composite   ▶
```

When saving a merged image you are presented with a dialog stating that the source image has attached data and asking you if you want the data saved as well. If your answer is affirmative then the data is saved separately in Raw format (see section Raw) so that it and the source image can be reloaded and merged together again.
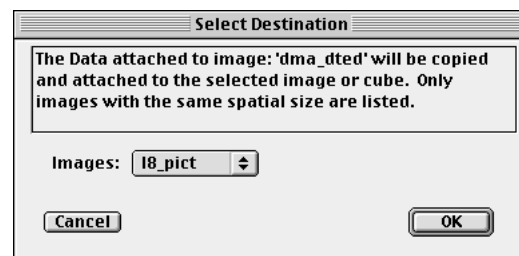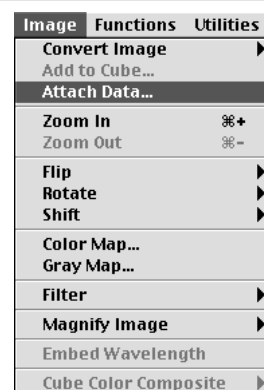
```
╔══════════ Select Destination ══════════╗
║ The Data attached to image: 'dma_dted' will be copied  ║
║ and attached to the selected image or cube.  Only      ║
║ images with the same spatial size are listed.          ║
║                                                        ║
║   Images:  [ l8_pict   ⬍ ]                             ║
║                                                        ║
║   [ Cancel ]                    [   OK   ]             ║
╚═══════════════════════════════════════════╝
```

## Statistical Measures

Functions->Statistics contains two methods for computing the linear correlation between 2 or more 8 bit images. The first of these, Pair Correlations..., allows you to select any two 8 bit images from popup menus and then calculates their correlation or covariance as a function of pixel and line offsets. The results are listed in a 5 by 5 matrix showing the computed values for each given offset. If either image was loaded with Retain source... checked then the actual data will be used in the computations. The Options... button permits changing the initial offset between the 2 images and the pixel and line shifts per iteration. If image 1 has a selection rectangle then the correlation computation can be confined to the selection by checking the Use Image 1 Selection Box. The check box will only be enabled if there is a selection rectangle on image 1. Pairs Correlations can be used to determine how good the registration is between various bands of a multiband image. The more symmetric the fall off from the zero offset the better the registration. Each computation is generated by selecting the Update button and may be aborted by pressing the command period (.) combination.
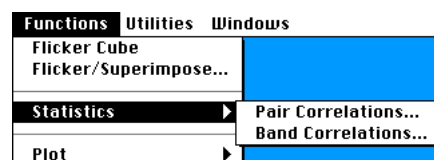
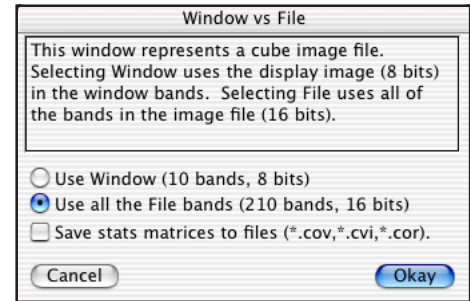*The linear cross correlation between two bands of a multiband data set using the default values for offset and delta is shown at the right. The corresponding covariance between the two bands is shown below.*
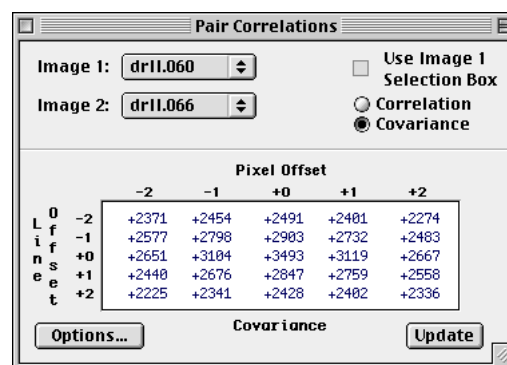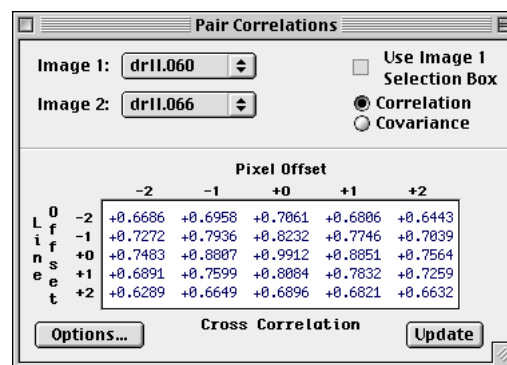
The options dialog window specifies the initial pixel and line offset as well as their respective steps for each iteration.
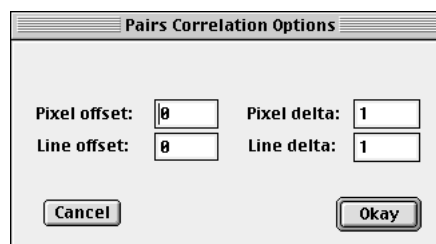
The second type of correlation in HyperCube, Band Correlations, is restricted to multiband images. It calculates various combinations of linear correlation coefficient, covarianc, variance and mean. You are presented with the dialog at the right giving you the choice of using either the window image data already in memory or the file image data. The result is shown graphically with a scrollable list window. Clicking on an item in the scrolled list highlights it on the graph. You can spot which bands don't correlate well with the others due to atmospheric phenomena or sensor errors. The radio buttons determine the type of comparison and the statistics. Sequential correlates each band with the next band, i.e. band 1 to band 2, band 2 to band 3 and so on. Specific allows you to select any band from the Band popup menu and compare it with all of the others. Shown here are the results of correlation applied to a 210 band image. The vertical marker in each graph corresponds to the highlighted values in the lists. The list values may be saved as text by choosing File Menu->Save when the dialog window is front most or copied, (Edit->Copy), to the clipboard and subsequently pasted as columnar data into a spreadsheet application such as Excel™. The correlation and covariance matrices may be saved to a file in ASCII format which can subsequently be opened in Excel™ (see Apply Transformation for details).
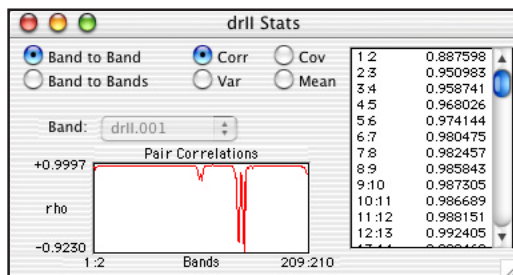
The sequential band to band correlation plot showing the absence of correlation within the water absorption portion of the spectrum. Generated from file data.
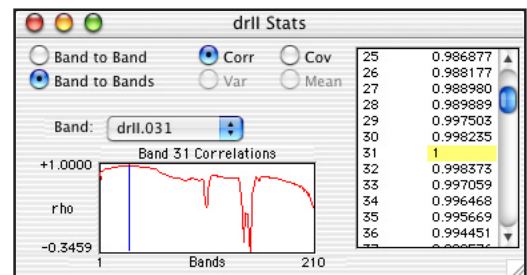

Same multiband data with band 31 compared with all of the bands within the cube including itself. Clicking a list item high lights its position within the graph.

If 32 or more bands are used then both the covariance and correlation matrices generated in the statistics are converted into images that can be further analysed by performing profile plots, histograms and examining values via the cursor and Info window.


The matching Covariance *and* Correlation *images corresponding to the statistics plots shown above. The image size is the number of bands and pixels represent floating point data, large values for* Covariance *and -1 to +1 values for* Correlation.
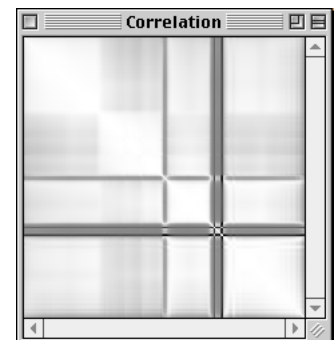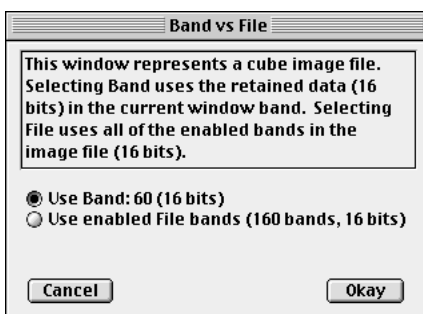
Histograms

A histogram of the distribution of the pixel values for any single or multiband image can be generated for the front window by selecting Functions->Plots->Histogram. If a selection rectangle is present the histogram is confined to those pixels within the rectangle. If the band list for a multiband image has any bands disabled then those bands are ignored in computing the histogram. If the window is a cube then you will be presented with the dialog shown below allowing you to choose either a specific band in memory or the file data. Also, see section Pixel Exclusion.
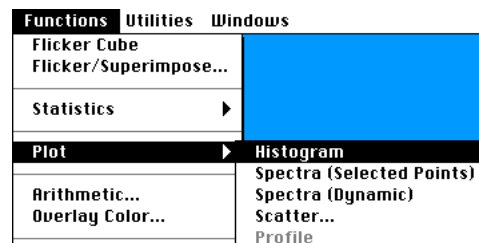
A true color image produces three separate histograms:, one for each of the three color components: red, green and blue.   If the window is a color map overlay then the output is a list of the frequency count of the classes (see section Image Products).
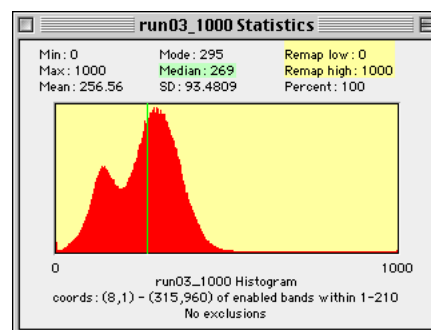
If the image was loaded with Retain source... then the histogram of the underlying data as well as the display gray values will be generated.
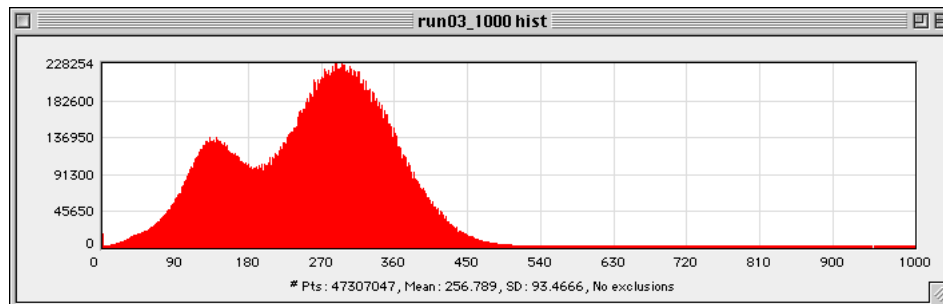

Cube source selection dialog.
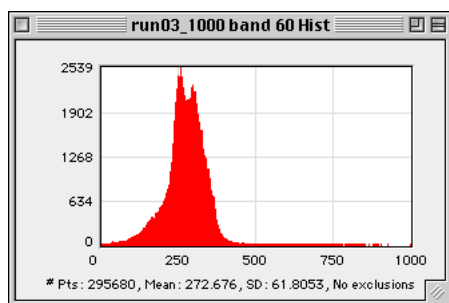

The histogram of just band 60.

All 160 enabled bands showing remapping limits.

The histogram of 160 enabled file bands. Can be resized or a selection magnified.

Pixel Exclusion

For display purposes and to yield better histogram plots you can select a value or range of values to be ignored (excluded from computation) when generating certain statistics. This option permits cube image pixels to be better mapped for display or to see details in a histogram that would be overshadowed by a too populous bin (e.g. a lot of zeros). Menu Applications->Shaded Relief also does not use excluded pixels in computing estimates for its initial x and y slope scaling but does use all pixels in generating the output image (see section Applications). Excluded values do not affect analytical operations such as Principal Components, Classifications and Image Arithmetic,
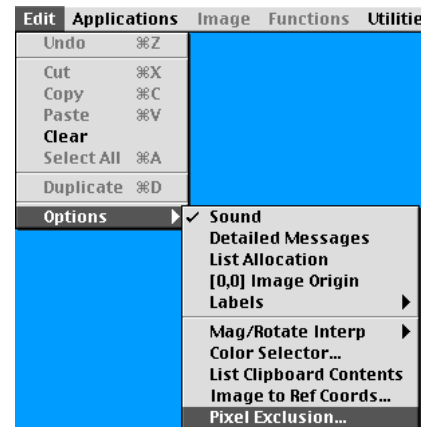
The following dialog is presented by selecting menu Edit->Options->Pixel Exclusion. Enabled exclusions are global, i.e., they are in effect for all images until they are disabled.

The dialog shown at right will cause all pixels in the closed interval [-153,0] to be excluded when generating Histograms *and determining the display remapping necessary when loading an image cube. This will also apply to the initial slope estimates in menu* Applications->Shaded Relief.
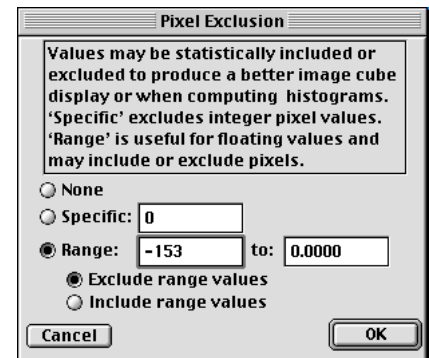
Scatter Plot

Choosing Scatter... from the Functions->Plot submenu brings up the image selection dialog window shown below. After selecting two 8 bit images, which may differ in size, a two dimensional scatter plot is generated with the origin, point [0,0], at the lower left. A plot point is derived by using image one's 8 bit pixel value as the y coordinate and image two's pixel value as the x coordinate as the images are scanned in unison. Therefore the output image will always be 256 by 256 and mostly black. Each time there is a multiple hit, i.e. the inputs generate an x,y already plotted, the color (or gray value) for that point is modified from blue (black) towards red (white). If both selections are the same image then the output image consists of a 45 degree broken line of colored dots from the lower left to the upper right. After the output image is generated you may select other color tables from the menu string Image>Convert Image>Color Table to view different encoding schemes.

By selecting the scatter plot and moving the cursor over it the corresponding pixels in the two input images will be overlaid with red dots. To force their gray scale restoration choose the first color table (linear ramp) as described in the previous paragraph.

The initial coding method is determined by these check boxes and can be later modified by choosing a different color table.
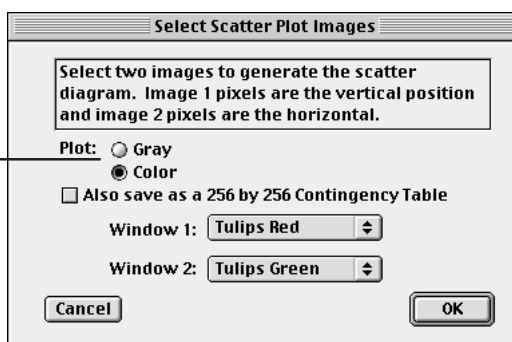
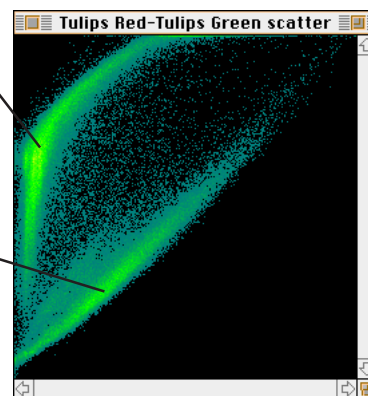*All currently open 8 bit gray windows, regardless of size, are listed in the popup menus.*
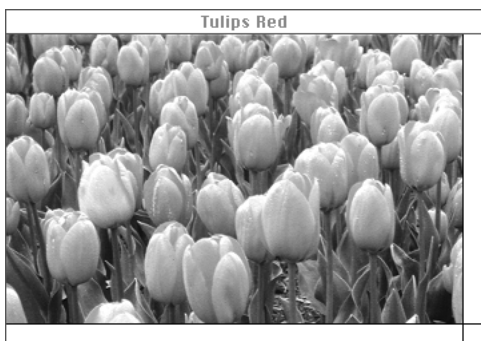
Red Tulips component.

Points in this region correspond to picture elements where red is the dominant color.

The diagonal band maps to values of similar intensity in the two images.

Green Tulips component.

The red and green component images of a true color image were used to generate this scatter plot image. The red image pixels are the vertical coordinate and the green pixels the horizontal coordinate. The red tulip flower tops form the cluster in the upper left. As the cursor is moved over the plot the inverse transformation shows which pixels in the source images contributed to this point.
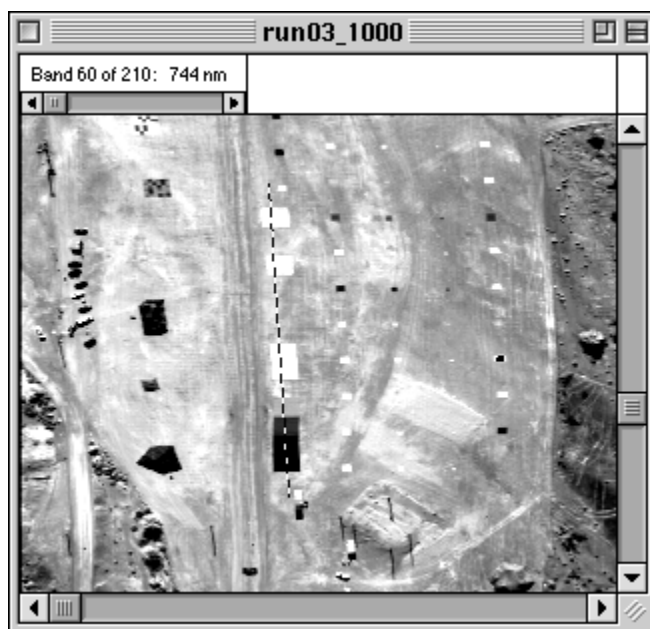
Line Profile Plot

A density value plot of an image can be generated by first delineating a line and then selecting the Function->Plot->Profile menu. The line is specified by positioning the cursor on one endpoint, depressing the control key (previous versions used the shift key) and clicking and dragging the mouse to the other endpoint. Upon releasing the mouse a marque-like line will remain. Repeat the procedure to select a different line. Control-click (control-double-click in the Windows version) the mouse without moving it to clear the line or choose the Edit->Clear menu.
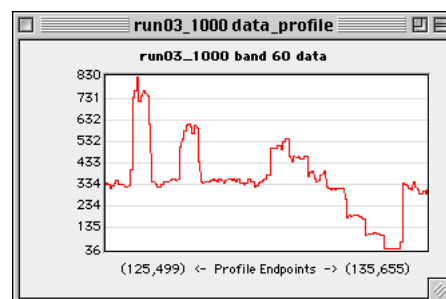
If an image is color, including pseudo color, then three separate profile plots will be generated for the red, green and blue components respectively. Each plot is resizeable and includes the coordinates of the endpoints such that the left coordinate is the start and the right coordinate the ending point. Overviews show the properly scaled coordinates. The vertical coordinates are expressed as either gray values (always 0 thru 255) or in data values ranging from the minimum to the maximum data within the profile line (see section Raw, Retain source... checkbox).

The profile plot window is an independent window that can be printed or copied to the clipboard, or duplicated and saved. To save a profile as an image activate its window and select menu Edit->Duplicate. Then save the duplicate as a normal image. The actual profile values can be saved as a text file by selecting menu File->Save. The file will consist of 1024 lines of x,y,z triplets regardless of the length of the profile line. Only the x and y are interpolated, each z value is exact. The x,y,z triplets are also copied to the clipboard when performing Edit->Copy. Moving the cursor over a profile plot line will display the corresponding image x,y,z values in the Info window if enabled (see section Info Window).

Profile values resulting when the Retained source data *option was chosen when loaded.*

A profile line drawn over band 60 of a hyperspectral cube with 16 bit pixels. The plots at right represent the effect of the Retain source data *in memory option when loading. See section* Raw.
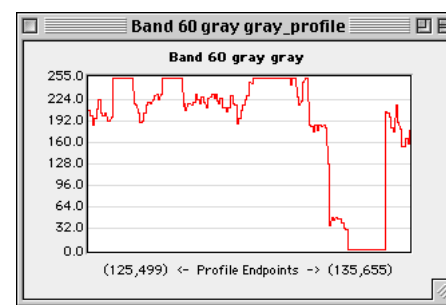
Same profile values without Retain source... *option when loaded.*

Image Arithmetic

HyperCube allows you to enter general "C" type arithmetic expressions where the operands are images and constants. The result is an image whose size equals the minimum spatial size and maximum pixel depth of the individual image operands. Since the arithmetic generally produces values outside of [0,255] you can specify how to remap the resultant display image. All arithmetic is performed to full precision and remains attached to the output image. If an expression contains a color image the output will be a color image limited to the display depth of 8 bits per color (see examples).

Menu Functions->Arithmetic... opens the dialog window shown below. The popup menu contains a list of all of the current images that you can use to form an expression and is in order of creation. Each image name is preceded by the letter "s" and a number. These symbols are used to form the expression instead of using the explicit image name. The expression itself will be used as the resultant image name and it too will be added to the image list. The expression is typed into the upper edit box and must conform to the "C" syntax. Listed below are the operators and functions recognized by HyperCube.

(,)	left and right parenthesis	==	equality
+, -, *, /	arithmetic operators	!=	inequality
%	modulo	>=	greater than or equal
&,|	bit-wise and, or	<=	less than or equal
^	bit-wise exclusive or	>	greater than
<<,>>	bit-wise shift left, right	<	less than
+,-	unary signs	pi	constant (3.14159...)

```
fabs, sqrt, pow, exp, log, log10
sin, cos, tan, asin, acos, atan, atan2
shift_left, shift_right, shift_up, shift_down - see examples
slopex, slopey, slopexy, gray - see examples
```

The relational operators (right column above) produce a mask image consisting of 1's where the pixel by pixel relation is true and 0's where it is false. For example: the expression s1==s2 creates a mask with 1 for each pixel of s1 and s2 being equal and zero otherwise. The expression: (s1==s2)*s1 shows those pixels of s1 that match s2, whereas (s1!=s2)*s2 shows those pixels of s2 that differ from s1. See more examples later.

The Image Arithmetic *dialog window displaying a list of images that you can use as operands in an expression. The example shown at the right generates an image mask representing the areas that have an x and y slope magnitude greater than 1.*
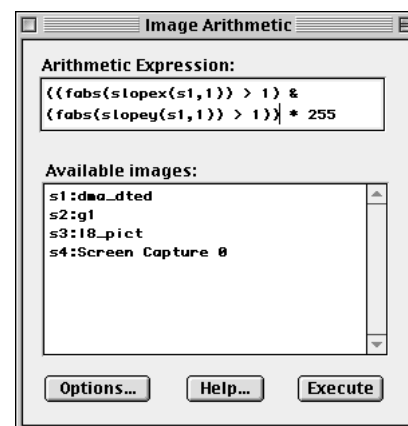
In addition, a new constant type operand is available that consists of a triplet set of numbers represented as {constant,constant,constant}. When used in an expression such as s1*{.95,1.3,.6} the red component of s1 is multiplied by .95, the green component by 1.3 and the blue component by .6. You can use the triplet constant anywhere that a single constant is valid. It will force the pixel depth to true color.

All image operands are converted to floating point before arithmetic is performed. The result of the expression must then be converted back to 8 bits/plane to display an image (the underlying data remains the results of the expression). The Options... button allows you to determine how the display pixels generated by the expression are remapped back to the [0,255] range. If the Remap check box is unchecked then the expression is truncated (but not the underlying data) such that values above 255 are set to 255 and values less than 0 are set to 0. This is useful when the expression only contains bit-wise operators. Generally, expressions produce pixels well out-of-range. In this case select the Remap check box and choose the min & max radio button, then the expression range [min,max] will be linearly remapped to [0,255]. For expressions that produce a wide non uniform range of values choose the 3 standard deviations radio button. Then the value of the expression is remapped so that values in the range [expression_mean - 3*sd, expression_mean + 3*sd] are linearly transformed to [0,255]. Values outside of this range are set to 0 and 255 respectively. The Do each color separately checkbox specifies whether one set of statistics is applied globally for all three colors or whether three sets of statistics are separately used to remap each color independently of the others. Excude zeros ignores any zero pixels when remapping.

Set divide... defines the result of any zero divisors that may occur (almost a certain consequence of using the divide operator). The 3 possibilities set the resultant pixel to zero, one or 255. The default is zero but the other 2 choices should also be tried especially if remap between min & max is set.

Expressions are evaluated from left to right unless parentheses force the order of execution. An expression can consist only of constants in which case the output image is a uniform color. All 8 bit pseudo color images are converted to their respective true color values during computation. The result remains true color, it is not reconverted to pseudo color. Any expression containing only gray results in a gray output image at full precision. Any expression containing color produces a color output image at 8 bits per color.

The Arithmetic Options dialog allows you to generate a sequence of outputs that are assembled into a "synthetic" cube where each plane is the result of one evaluation of the sequence. To use this option check Generate a sequence, and enter the number of ouputs and range that will be linearly substituted for the variable 't' in the expresion. For example: the arithmetic expression **slopex(s1,t)** with the options shown will generate a cube consisting of 10 planes where each plane is the slope in the x direction over an interval of 't', i.e., 1,2,3,...,10. The variable 't' is not limited to integer values and may be substituted anywhere in the expression that the substituion produces correct syntax.
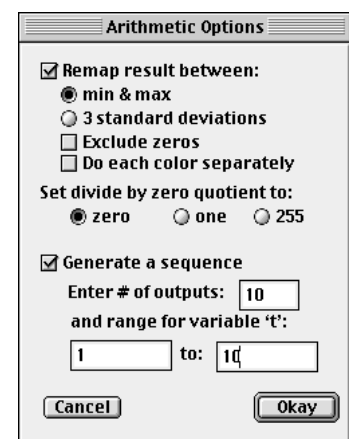
Examples:

where: `s1` is a data image; `s2`,`s3`,`s4` are bands of a cube and `s5` is the shaded relief of `s1`

`((s1>1000)&(s1<1200))*s1` Keep `s1` values between 1000 and 1200, zero rest.

`log(s1)` Should also choose a remapping option.

`atan2((s2-s3),(s2-s4))` Also try various remappings.

`shift_left(s1,3)` Shift the `s1` image 3 pixels to the left with wraparound.

`shift_up(s1,2)` Shift the `s1` image up 2 lines with wraparound.

`slopex(s1,2)` Generate the slope of image `s1` over an interval of plus and minus 2 pixels in the x direction.
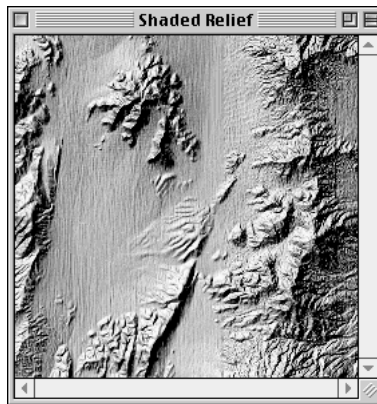
`gray(s5)` Use the display gray values of s5 instead of any attached data.

`(s1>1200)*{32,0,0}+gray(s5)` Generate a red tint image of all regions of `s5` that are greater than 1200.
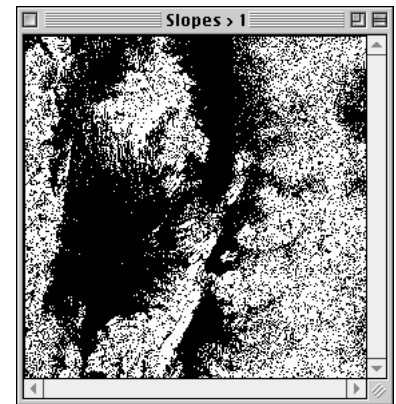
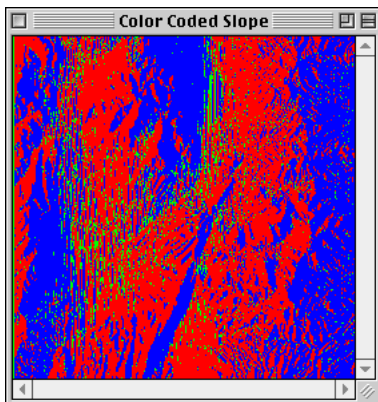`(fabs(slopex(s1,1))>1&fabs(slopey(s1,1))>1)*255` A mask of areas with x and y slope magnitude greater than 1.

A data source image.

Corresponding shaded relief generated via menu Applications->Shaded Relief.

The mask image resulting from the last expression in the examples.
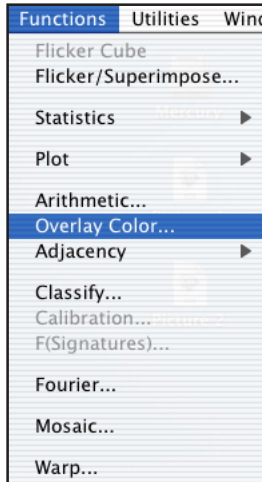

This image of the above source image is the result of applying the the following arithmetic expression:
`(slopex(s1,1) > 0)*{255,0,0} +`
`(slopex(s1,1) < 0)*{0,0,255} +`
`(slopex(s1,1) == 0)*{0,255,0}`
*so that positive x slopes are colored red, negative x slopes blue and zero x slopes green. Expressions may be copied and pasted to and from a text window (*File->New*) and then saved.*

Functions   Utilities   Wind

- Flicker Cube
- Flicker/Superimpose...
- Statistics ▶
- Plot ▶
- Arithmetic...
- Overlay Color...
- Adjacency ▶
- Classify...
- Calibration...
- F(Signatures)...
- Fourier...
- Mosaic...
- Warp...

A special image function is provided to allow overlaying one image on top of another. This could be accomplished by splitting the images into red, green and blues planes followed by various logical operators previously described in Image Arithmetic and then re-combining the resulting planes. The Functions->Overlay Color... menu makes this much easier: you select the two images and the program will construct a third true color image that is the result of overlaying image 1 upon image 2. The first image is the mask or top and the second image is the base or bottom. Every black pixel in the mask becomes clear when placed upon the base thus allowing the base image to show through at that pixel. Every non-black pixel in the mask overwrites the base image. The spatial sizes and pixel depths of the source images do not have to be the same, the output image will be the size of the overlap and true color. Psuedo color and gray scale are automatically accounted for. Note that the mask criteria is based upon color, in this case black, and not on a zero index if the mask happens to be a pseudo color image.

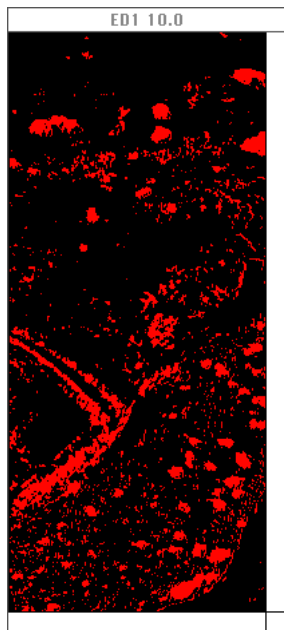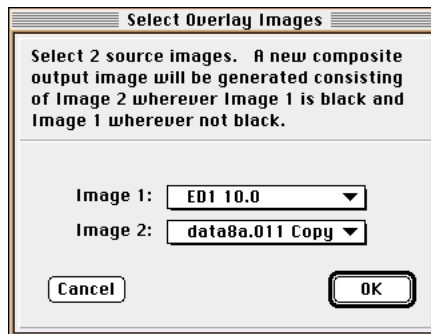Choosing Overlay Color... from the Functions menu brings up the following dialog:

Select Overlay Images

Select 2 source images. A new composite output image will be generated consisting of Image 2 wherever Image 1 is black and Image 1 wherever not black.

Image 1:   ED1 10.0 ▼

Image 2:   data8a.011 Copy ▼

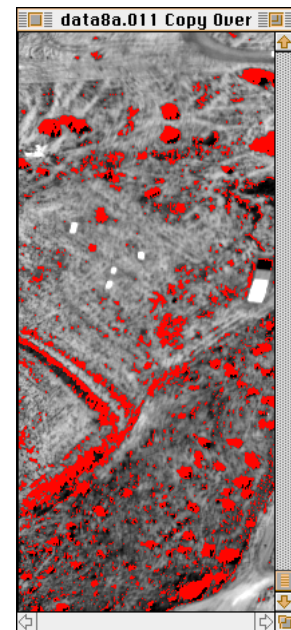[Cancel]   [OK]

ED1 10.0   data8a.011 Copy   data8a.011 Copy Over

*Image 1 (mask)*   *Image 2 (base)*   *Color Overlay*

Adjacency

This image function can be used to delineate where two or more pixels (classes) have the same value within a selected spatial neighborhood. Four variations are available: the first, Images, cross compares the pixels in two <u>different</u> images, the second, Colors, uses a <u>single</u> image in the comparison and the third and fourth, Classes and Singularities, pertain to spectral classifications. The idea is the same for all: generate an output image consisting of a combination of input image pixels that match some criteria anywhere within a specified radial distance of each other. The neighborhood can range from zero to some arbitrary radius.
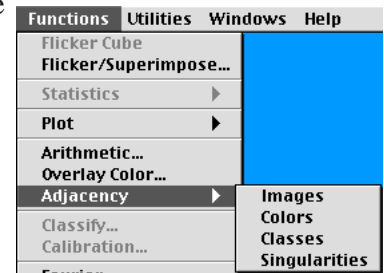
Adjacency->Images is a generalization of equality. Given two images it sequentially compares a pixel of the first image with the neighborhood surrounding the similarly located pixel in the second image. For example, a radial neighborhood of 1 is interpreted as comparing the source pixel with any of the 4 adjacent pixels in the second image (a radius of 1 is not large enough to include the diagonal neighbors). If Coincidence is checked (see below) then the central pixel is also included. A radius of 2 increases the valid comparison region to 12 or 13 pixels depending upon Coincidence. Fractional radii are allowed, e.g. a radius of 1.42 would include 8 or 9 pixels. If the radius is set to zero then the function reduces to equality. The output pixel can be set to just the source pixel, just the matching neighborhood pixels or, both as selected by the three radio buttons. The image examples below demonstrate how two spectral classification output masks are combined to produce a new mask in which the algorithms agree with each other allowing a spatial uncertainty of 1 pixel, e.g. where Vector Angle and Correlation Coefficient matched exactly (Coincidence set) <u>and</u> within 1 pixel of each other.
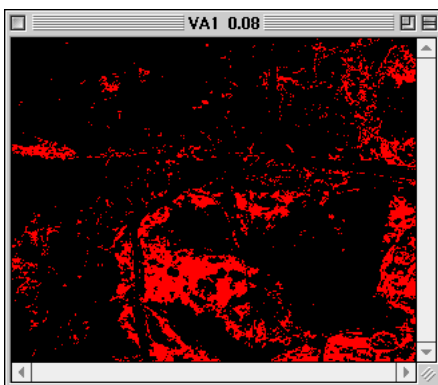

The two source spectral classification outputs used as input to the Adjacency *function producing the output shown at the lower right.*
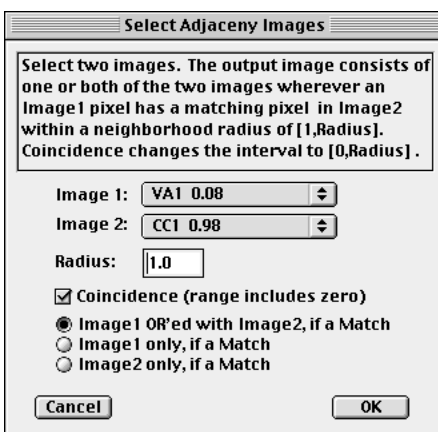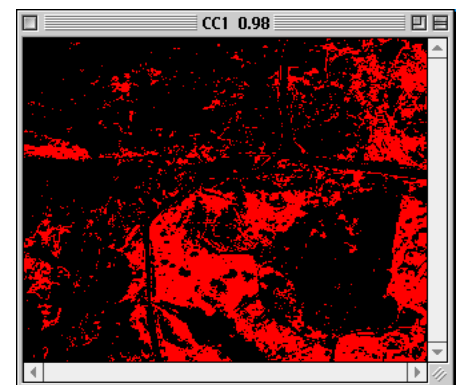

Dialog is set to produce an exact match as well a spatial mismatch of a one pixel radius.
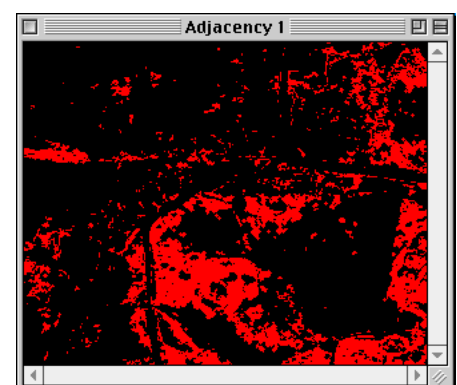
Adjacency->Colors operates on the front most image window and is used to extract pixels with two specific colors that are co-located within a selected radius. This operation could be done by first separating the source image into two masks of the given colors (see section Image to Mask) and then using Adjacency -> Images as above but Adjacency -> Colors greatly simplifies this. The image example below shows how a multiple spectral classification output can be transformed into an output showing the outline of everywhere one class is exactly adjacent to another class, e.g. grass and tree canopy demarcation. Again, the selected radius allows a considerable latitude in the equality range. The two colors to be compared are best chosen by first positioning the cursor over the source image and using the special command-option-C combination described in section Copy, Paste & Duplicate. The colors will then automatically be filled in when Adjacency ->Colors is selected. Note that checking Coincidence has an effect only when the two selected colors are identical in which case the resultant image will consist only of that extracted color (see section Image to Mask to accomplish the same thing).
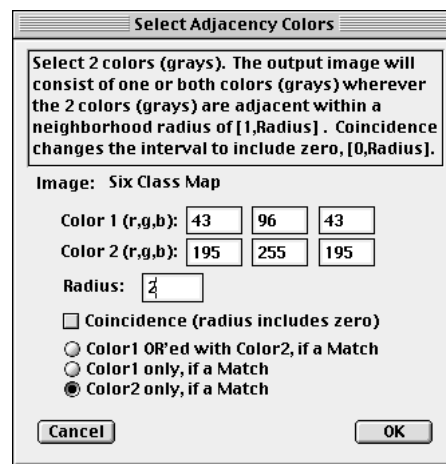

The dialog above was used on image Six Class Map to extract tree [43,96,43] regions that are adjacent within a 2 pixel radius to grass [195,255,195]. The output consists of only the grass border.
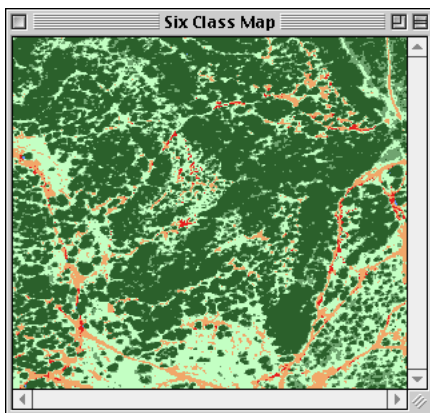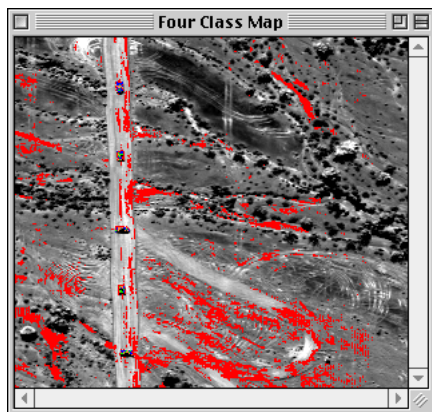
The third variation, Adjacency->Classes, is only available when the front most image is a classification overlay produced via the Classify function. This type of image displays the member class as the cursor is moved over it. The intent is to extract all of the areas within the overlay that have a common adjacency but instead of specifying colors you specify how many of the class members must be present within a certain radius. If you select less than the actual number of classes that the overlay possesses then this subset could and would change throughout the image. This permits generating a spectral classification using similar signatures from a library (again using Classify) and then entering how many of these signatures are needed (found) within a certain radius to reduce the ambiguity of false "hits". The output may be an image or a mask depending upon the Clear image background checkbox. The image appears where the criteria fails and some combination of the class member colors where it succeeds. If a mask then black appears where the criteria fails. The mask can be overlaid on the original classification source using Overlay Color in the Functions menu.

The following user dialog and examples illustrate how you can increase the accuracy of a spectral library classification.
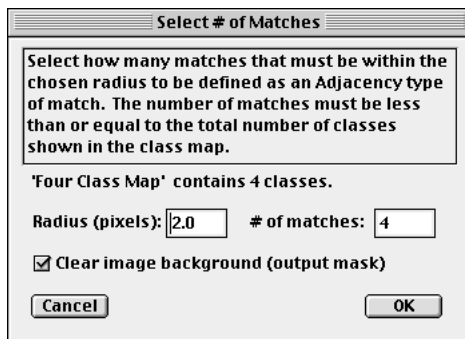
Class map image Four Class Map *consists of 4 classes. All four classes are used and searched within a 2 pixel radius to generate the output mask image shown below right.*

Vehicles

This classification overlay image was generated by using 4 signatures from a spectral library, each depicting some characteristic of a vehicle.
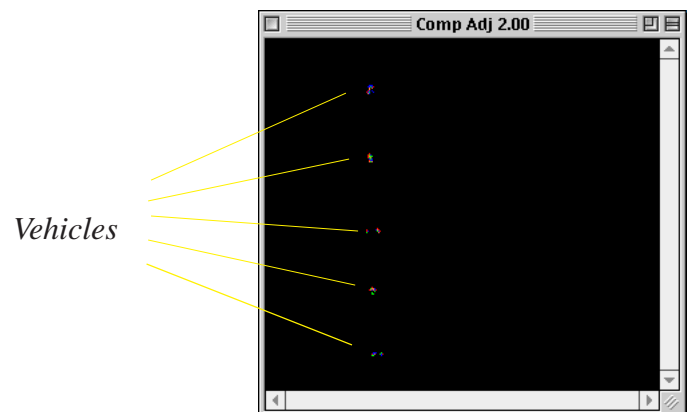
After applying Adjacency -> Classes *to the left hand image.*

The fourth variation, Adjacency->Singularities, is also only available when the front most image is a classification overlay produced via the Classify function. This type of image displays the member class as the cursor is moved over it. The idea here is to scan the image for all occurrences of a single class map color (feature) that is surrounded by a different feature or features within a specified pixel radius. Wherever this happens the singularity is replaced by the surrounding feature with the greatest abundance within the radius. If the single class feature is surrounded by imagery (gray) then no replacement is performed. The final output of this function is an almost identical class map with single pixel "noise" replaced by surrounding features. Since this modifies what the Classify function thought was correct you must carefully inspect the new class map to determine the validity of this operation. The best use of this function is to de-clutter class map images before converting them from raster to vector form (See section Class Map Editor).

Below are the user dialog and an example of applying this function.
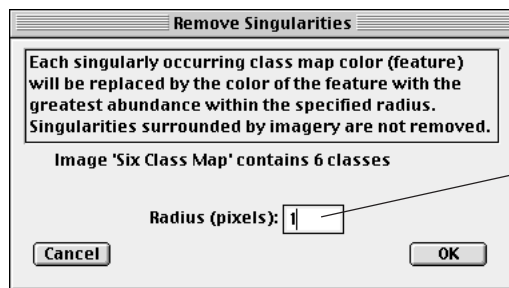
As the radius increases, the probability of finding a singularity greatly decreases. With a radius of 1, only the top, bottom, left and right pixels need to be different.
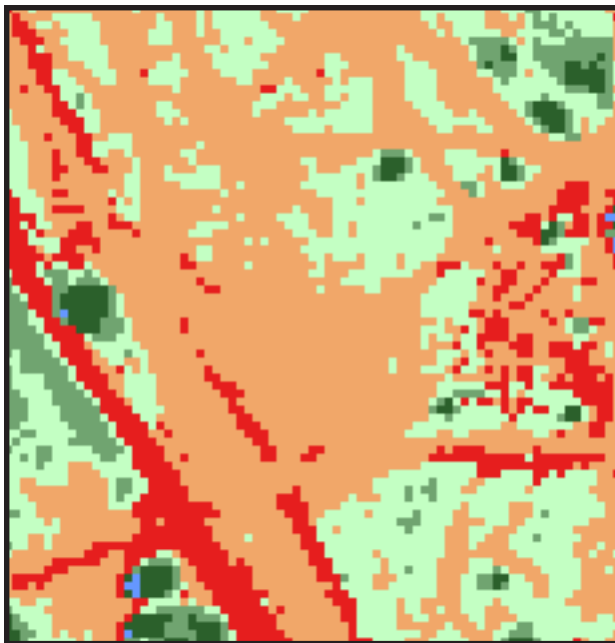
A three times magnification of a portion of a six class map as output from the Classify *function. There are 169 singularities within this subset.*
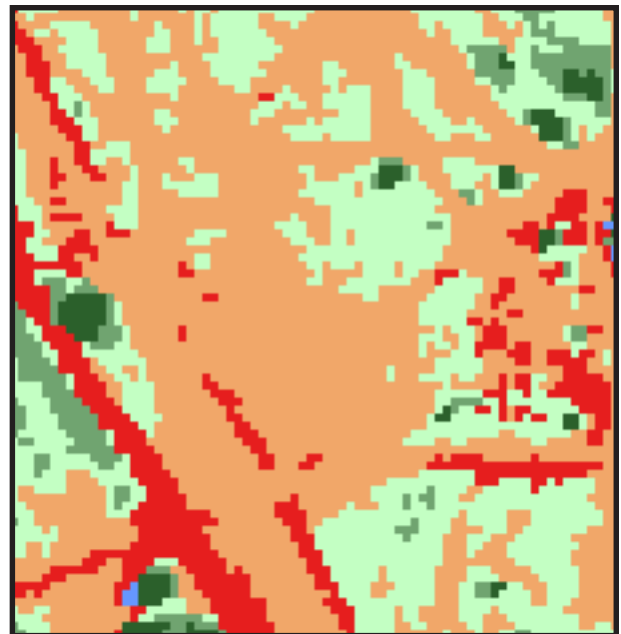
The same subset after removing each occurrence of a single class surrounded by different classes.

Fourier Operations

Frequency domain filtering is initiated by selecting the menu Functions->Fourier. The dialog presented allows you to interactively change various filter parameters and apply them to an image. The filter's characteristic curve is shown within the plot window as sliders are moved. The radii sliders represent the half magnitude point. The attenuation slider determines how sharp the filter is. All filtering uses the same scenario: select part or all of an image, click the FFT & PS button, select the filter type, move the sliders for the desired effect, and click the Inverse button to generate the filtered image. Fourier is an operator like gray mapping, it is unique to the front image window when it was selected. There can be several Fourier operators, each connected to its own image. If an image represents data (see Retain source... in sections: Raw, Multiband and TIFF) the transform is applied to the underlying data and the result is also a data type image.

An important visual key to interactive frequency domain filtering is the power spectrum (PS). It displays the relative amplitude of the frequencies comprising the image. The center of the power spectrum represents the lowest frequency with frequency increasing radially outward. HyperCube shows the power spectrum as an image with the slider's radii superimposed upon it. As you move the sliders the radii change indicating which frequencies will be affected for the type of filtering chosen.

Source image selection rectangle embedded in a 256x256 background.
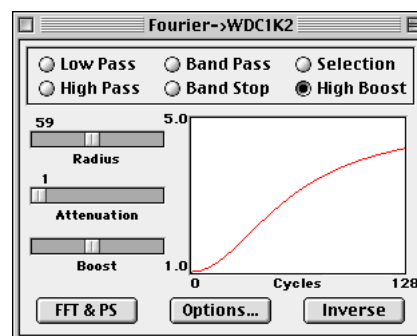
Fourier dialog with High Boost *filter chosen and applied to the transform of the image at the left.*
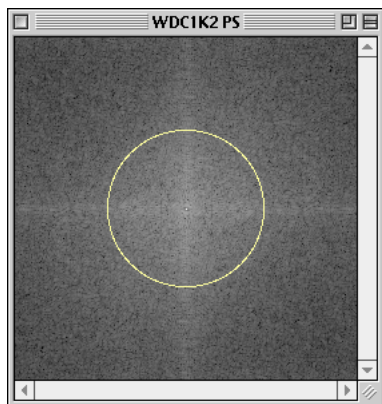
Power spectrum of the embedded image using log compression
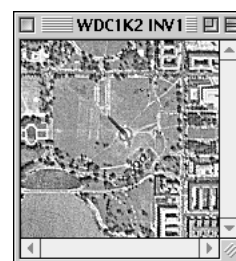
The resulting transformed image selection extracted from the embedded background.

Fourier contains five slider controlled filters and one user selection filter. These are defined as follows:

Low Pass   Attenuate frequencies above Radius 1. Use this slider and the Attenuation slider for desired response.

High Pass   Attenuate frequencies below Radius 1. Also use Attenuation slider to shape response curve. Because the average brightness is zeroed remapping (Options) must be chosen.

Band Pass   Pass only those frequencies between Radii 1 & 2. Radii sliders select pass band, Attenuation slider determines sharpness. Also requires remapping option.

Band Stop   Attenuate frequencies between Radii 1 & 2. The inverse of a pass band.

High Boost   Similar to High Pass. The boost varies from 1.1 to 9.9 and is controlled by the second radii slider (label changes to Boost). Remapping should be used for data type images.

Selection   Attenuate frequencies within a user selection rectangle. Manually draw a selection rectangle on the power spectrum image.

The image area to be filtered must be a power of two sized square. HyperCube accomplishes this by automatically embedding the source image or selection rectangle in the smallest power of two square image that will enclose it. The border is set to the mean of the image (selection) to minimize ringing in the inverse transform. All operations then proceed using the embedded image. The result will be extracted as a separate image and may be optionally pasted back into the source image. If the source is an 8 bit pseudo color image you will be directed to convert it to a true color copy using menu Image->Convert Image->Indexed to Color.
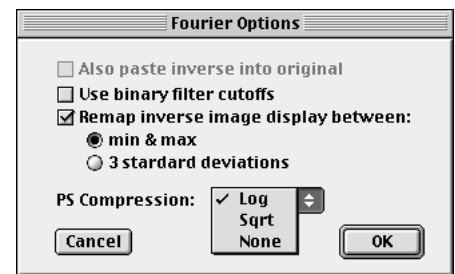
The Fourier Options dialog shown above presents additional controls. The top check box determines whether the inverse image also overwrites the original image patch. The second check box changes the filter characteristic from a tapered one to a vertical cutoff. The Remap check box works exactly as that in the Image Arithmetic section and is a must for high pass and band pass filters which zero out the fundamental frequency (image average). This is also especially true for any images that represent data (see Retain Source... checkbox in sections: Raw, Multiband and TIFF). The PS Compression popup is used to present a better display of the power spectrum.
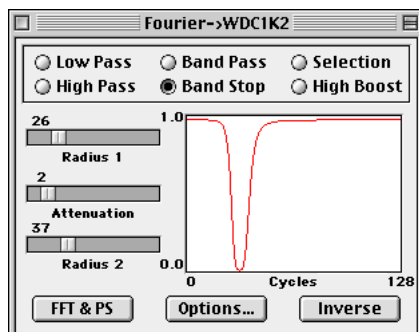
Typical characteristic curves for Band Stop, *left, and* Band Pass, *right,* Fourier *filtering.*
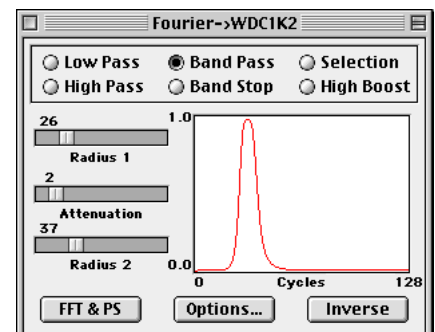
Mosaic

Two or more images can be fitted together to form a composite assemblage. The manner in which they are joined can be uncontrolled by selecting conjugate image points on each of the images or controlled by matching image points to a list of externally derived coordinates. The geometry used to force the joining ranges from a simple orthogonal transformation to a plane to plane projection. However, the transform must be the same for all of the images. Mosaic is more general than Warp (described in the next section) in that the end result is to stitch images to a common geometry for a greater view whereas warp's objective is to match the geometry of one image to another.

Choosing the Functions->Mosaic menu, a dialog will appear with several scrollable list areas. Only the Available Images list initially contains anything, namely all open images that are candidates for mosaicking. Click on each Available Images list name to highlight it and click the Add -> button to move the image name to the Mosaic Images list. Double clicking an Available Images name will also move it directly. The order of selecting images is not important. To remove an image from the Mosaic Images list click its name and click the Remove button or double click its name, it will be returned to the Available Images list. Removing an image after starting the control point selections will remove all control points. You are warned of this and are allowed to cancel the removal.
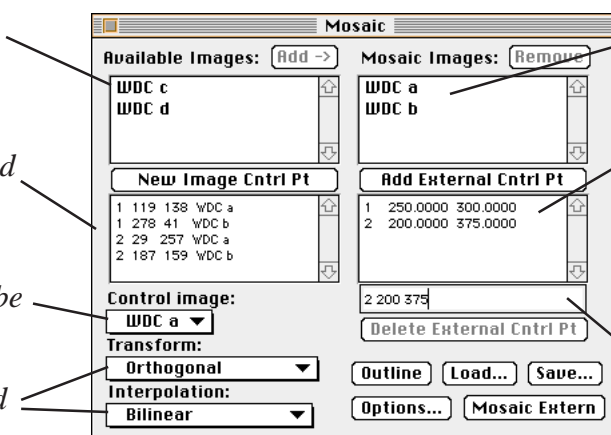
Once the mosaic list has been determined the program is ready to begin the control point selections. Position the cursor over any image and shift-click the mouse, a colored numbered reticle will be left there. Move to the conjugate position on another image and shift-click there. This can be repeated until each image contains the same matching control point. Subsequent attempts are disabled until the New Image Cntrl Pt button is clicked. This permits selections for the next control point number. Each control point must appear on at least 2 images before going to the next control point. New Image Cntrl Pt is automatically disabled until at least 2 images contain a given control point. Each control point does not have to appear on each image, only on at least two. As each control point is chosen it appears in the lower left scrollable list. A point may be deleted my clicking on it and choosing the Delete Image Cntrl Pt button. It will be deleted from each image it appears upon and the remaining points will be renumbered. The number of control points determines the degree of transformation (see section Warp) that will be allowed and specifically, the minimum number of control points appearing on two or more images. Therefore if two images have only 2 control points in common then only the Orthogonal transformation will be available regardless of how many common points may appear on other images.

Unselected available images. Data type images may also be mosaicked together.

Two control points and their coordinates that have been measured.

The image chosen to be the geometric control

Type of transform and interpolation.

List of images to form the mosaic.

*External control has been added so that the mosaic can be fitted to these control points.*

Edit box.

After all of the control points have been selected you can mosaic the images by clicking the Mosaic Images button or generate an outline showing how the images will be placed in the mosaic by clicking the Outline button. This generates a colored line drawing image that can be overlaid on top of the final mosaic using Overlay Color... in the Functions menu. The Control Image popup is used to select which one of the mosaic list images defines the geometry that the remaining images are conformed to. The control image shape stays the same and the other images are warped to its geometry using the selected Tranformation and type of Interpolation. There are many options that determine how the mosaic is performed and the most important one is whether to fit the images to the control image geometry or to a set of externally defined control points.

The previous description was based upon the control image geometry. Any of the mosaic images may be chosen as the control and it will appear "stationary" in the output. The biggest problem with mosaicking to a control image is the cascading of errors: e.g. Image 1 is the control and has points in common with image 2 which has points in common with image 3 but image 1 and 3 have no (or not enough) points in common. Then any errors between 1 & 2 and between 2 & 3 cause very obvious misfits between overlap boundaries because the global least squares fit is to the control image's geometry. A way to minimize the effect of poor control points is to choose a good image as the control image and then only mosaic it and one other image together, then save the resultant mosaic and restart a new mosaic operation using the added image as the new control image (the current mosaic will generally have black areas that would overwrite if it were chosen as the control image). This has the effect of breaking the geometric fit into regions (more degrees of freedom) and results in a more aesthetically pleasing final mosaic.

The alternative method of mosaicking is to fit each image to an externally defined geometry where every image control point has a corresponding two dimensional coordinate such as X and Y or latitude and longitude. The steps here are similar to the above. First, you visually identify and select each image control point for which there is a known external coordinate. Then enter each external control point by typing the control point number followed by the two ordinates in the edit box and click the Add External Cntrl Pt button. Repeat for each control point. The input order doesn't matter, the point number determines the correspondence. A point can be deleted by clicking it (highlighting) and choosing the Delete External Cntrl Pt button or edited by clicking it, correcting it, and choosing the Edit External Cntrl Pt (Add External Cntrl Pt automatically changes to Edit... when a point is highlighted). The final step is to choose the Options... button and in the subsequent dialog to select the Use external control checkbox. Note that by toggling this option a mosaic can be switched back and forth between image control and external control. See the later sections for all of the subtleties available with the various options.
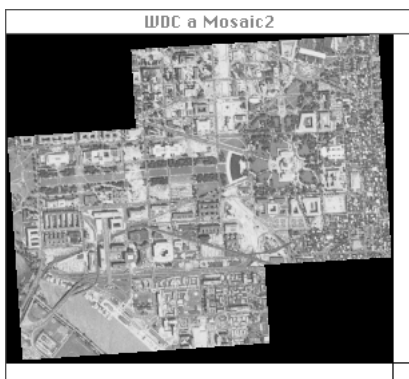
The result of mosaicking the two images in the previous dialog to external control is shown on the left along with the outline generated for the same pair to the right. Note each control point.
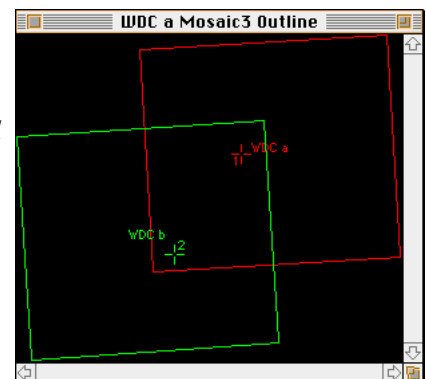
Listed below is a detailed description of each of the Mosaic window controls along with the possible options and their effect.

Add ->: Move the selected image from the Available images list to the Mosaic images list. Double-clicking does the same.

Remove: Move a selected Mosaic image back to the Available list. Also may double-click.

New Image Cntrl Pt: Used to step to the next control point, is disabled until the current point has been selected on at least two images.

Add External Cntrl Pt: After entering the external control point number and two ordinate values in the edit box this button will add the point to the external control list. When an external control point is selected in the list this button changes to Edit External Cntrl Pt and the selected point appears in the edit box.

Delete External Cntrl Pt: Any external or image control point may be deleted by selecting it and clicking this button. It toggles to Delete Image Cntrl Pt when an image control point is selected.

Control Image Popup: Selects which of the mosaic images will be the control geometry. If Use external control in the options dialog is selected this selection has no geometric significance. In either case the control image will always be the top most image in the mosaic, i.e. it will overlay the others in common overlaps. See Options.

Transform Popup: The type of geometric transformation. See section Warp.

Interpolation Popup: What type of interpolation to use in the resampling. See Warp.

Outline: Generates an output image the same size as the mosaic images but only shows the outlines of the individual mosaic images. Becomes enabled when a mosiac is possible. Also see Options.

Load...: Load a previously saved set of external coordinates.

Save...: Save the current, if any, set of external control points as an ASCII file.

Options...: Bring up the Mosaic Options dialog. See below.

Mosaic Images: Execute the mosaic operation. If Use external control is set in the Options dialog then this button changes to Mosaic Extern.

Selecting the Options... button displays the following user dialog.

The external control points shown here were implicitly determined after the previously shown example was mosaicked. Because Auto size *was selected the limits were computed by the program so that they encompassed the entire mosaic. If* Use external limits *is selected then the mosaic will be clipped to that exact area of* X,Y min and max.

Each option and its effect is listed below.

Average pixels...: Where the mosaic forms overlaps the control image's pixel takes precedence followed by the order of the Mosaic images list. When this option is checked all of the images within an overlap will be averaged together to form the output pixel.

Display mini...: When checked the Outline button will generate a miniature instead of a full scale line drawing. The mini cannot be correctly overlaid with the Mosaic output.

Use external control: The mosaic will be constrained (warped) to the corresponding points in the external control point list in the Mosaic dialog.

Auto size...: If selected then the limits of the external control point geometry will be determined by the limits of the final mosaic i.e. by computing the bounds of all of the corner coordinates of the individual mosaic images. This is mutually exclusive with Use external limits... below.

User external...: You may input the geometric bounds limits of the external control. The mosaic will be confined in output size to these limits. Note, if Auto size is first selected and a mosaic performed then returning to the Options dialog will show the program computed limits that enclose the entire mosaic. These can then be edited to customize the output after selecting Use external.

Warp

The geometry of one image can be empirically modified to fit the geometry of another image. You can force two completely different types of imagery, such as optical and radar, into registration, combine them and thus produce a new product. The registration points, the pixel resampling and the type of transformation used are all selectable. Warp is the Hyper-Cube function that accomplishes this.

After choosing the Functions->Warp menu, you select two images from the dialog popup list of images. All single images, including 8 bit, true color and DTED/Data are permissible. Using the mouse control points are selected alternately between the images as follows: position the cursor on control point 1 of image 1 and click the mouse while holding down the shift key, then position the cursor on the corresponding point 1 of image 2 and shift-click, repeat until all control points have been selected. Each pair of points leaves a cross hair mark behind, usually red on image 1 and green on image 2. The coordinate pairs are listed in the Warp dialog window where they can be edited or deleted. Either image may be zoomed to help position the cursor. The number of points needed depends upon the transformation, higher order requires more evenly distributed points.

The result of warping is a new image having the same size as the destination (->) image and depth of the <u>source</u> image. Depending on the transformation there may be pixels in the output image that have no correspondence in the source image, these will be set to Off image value. Transformations are performed in a least squares fit manner to distribute the error across the control points. The Residual button shows the mean error of the fit without producing an output image.

Shown here is the Warp dialog after selecting 40 corresponding points between a color composite and a control band.

Select images

Directs the user

Eight types available

Pixel resampling

Direction of warp

Warp an entire loaded cube image.

Click in list to select, then edit or delete.

Edit field

Or data value

Execute warp

Residuals and coefficients for the above example when the Residual *button is clicked.*

The transformation selected has a great affect on the shape of the output image. The nonlinear transforms are very sensitive to the number of control points and their positions. Some of the transforms along with their arithmetic are defined as:

Orthogonal:
```
xs =  a*x + b*y + c        Affine:    xs =  a*x + b*y + c
ys = -b*x + a*y + d                   ys =  d*x + e*y + f
```

Non-linear(s):
```
xs =  a*x + b*y + c*x*x + d*y*y + e*x*y + f*x*x*x + g
ys =  h*x + i*y + j*x*x + k*y*y + l*x*y + m*y*y*y + n
```

Projective:
```
xs = (a*x + b*y + c)/(d*x + e*y + 1)
ys = (f*x + g*y + h)/(d*x + e*y + 1)
```

The orthogonal transform preserves shape, allowing only rotations and a single scale change. Two points are required to define it. The second, affine, is a linear transform that preserves parallel lines and is a generalization of orthogonal. It will transform a square into a rotated parallelogram and is the most useful transform for registering two similar images. The nonlinear transforms include cross product and second degree terms and produce curved lines. The examples below show how sensitive they are to a slight change in the control coordinates. Affine requires 3 control points, non-linear(s): (4,4,6,8) and projective: (5).

Three methods of pixel interpolation are possible: nearest neighbor, bilinear and cubic resampling. Nearest neighbor is the fastest, it simply takes the closest source pixel and uses it as the output. That is important if multiband imagery is warped and then used in a classification as no new intermediate "fringe" classes are created. Bilinear interpolation uses 4 surrounding pixels to form a positive weighted average. Cubic also uses weights but some weights are negative. This produces a slight sharpening in the output image and appears superior to bilinear for natural images but produces ringing in artificial images that contain discontinuous gray values. For scale changes between .5 and 2 bilinear and cubic interpolation are almost indistinguishable.

The figures below demonstrate interpolation and transformation of the letter "E" warped to a duplicate of itself. The control points were purposely chosen to be slightly different. In the last figure a control point was edited to increase the mis-registration in order to show how great a change occurs in the nonlinear transformation.
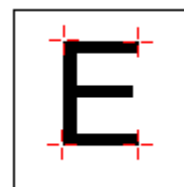
Orthogonal, nearest neighbor.   *Orthogonal, bilinear.*   *Affine, bilinear.*

Nonlinear, bilinear using original control points.

*Nonlinear, bilinear after editing one control point.*

Band List

Every multiband image has an associated band list that specifies the wavelength and status of each band. This is necessary in order to correctly plot spectra and perform library searches and comparisons. You can view a band list by selecting menu Windows->Show Band List . A resizeable window with a scrollable list of 4 columns and several buttons will appear. The columns display the band number, wavelength, band width and status for each band. Except for band number, the items are editable by: clicking them, typing a new value in the edit field and clicking the Edit button. Double clicking the band number (column 1) will automatically load that band as a single image not connected with its cube parent using the cube's remapping stats. You must use Load Selection in the Windows version in order to load a specific band.

The band list is the window display of the *.wvl file described in the Wavelength File section. If a *.wvl (or older .wav) file exists when the multiband image was loaded then HyperCube converts and loads its contents into the band list. The *.wvl file may also be loaded via the Load wav... button. By choosing Show Band List followed by a combination of editing and the Options... button, the band list can be modified as desired.   It is then saved as a new *.wvl file via the Save wav... button. Using File->Save or Edit->Copy will include all 4 columns, see section Copying and Sending Text to another Application.

Band list window controls have the following definitions:

List column: 1 Sequential band number, same as image cube band numbering. If the band wasn't loaded, i.e. only a subset of the full cube, an asterisk appears after that band number. Double clicking loads this band as a stand-alone image.

2 Band center wavelength in nanometers (must be increasing).

3 Band width in nanometers.

4 Status, if "disabled" (no check mark) then the band is ignored in spectral classifications, library searches and histograms. It will be highlighted in red on spectral plots.
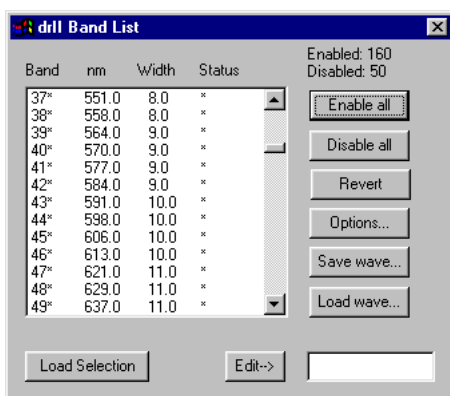
The Windows *version has an explicit* Load Selection *button, instead of double clicking.*
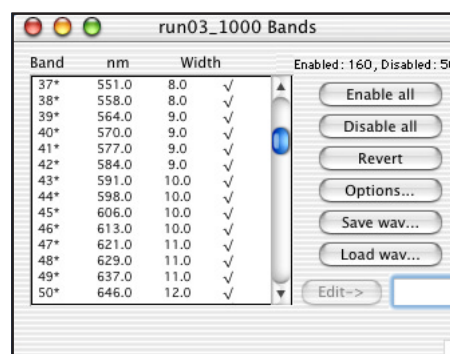
Mac *version band list showing the same selection area.*

Enable all	Set the status of all bands to enabled (check mark). Toggle individually by clicking on Stat (4) column entry.
Disable all	Set the status of all bands to "disabled". Also toggle individually as above.
Revert	Restore the band list to the last loaded wave length file.
Options...	Display the Band List Options dialog. Sets up various distributions of wavelengths that can be edited. See details below.
Save wav...	Save band list as *.wvl file.
Load wav...	Display a standard open file dialog and load the *.wvl file.

In combination with the Band List Revert button, Band List Options allows you to define a variety of enabled bands for experimentation. There are 5 options, the first 2 change the wavelengths and the last 3 enable or disable bands. However, the 3 Enable/Disable options will not <u>enable</u> any band that is currently disabled in the Band List.

Start & end *is used to set a uniform sequence of wavelenghts.* Delta *is computed.*

Start & delta *is similar to* start & end *with a user defined* delta.

Only *those wavelengths within the interval values defined in* Image -> Cube Color Composite -> Definitions *are enabled.*

The complement of the above option, i.e., the Color Comp Defs *are disabled.*

The last of the 3 Enable/Disable *options: Only enable every n'th band that was already enabled, e.g., a value of 2 enables every other band currently enabled.*

Band List Options

Linearize wavelengths:
- ● Use start & end
- ○ Use start & delta

Starting wavelength:	397.0
Ending wavelength:	2495.0
Wavelength delta:	N/A
Band width:	1.50 times delta

Enable/Disable specific wavelengths:
- ○ Use Color Composite Defs
- ○ Exclude Color Composite Defs
- ○ Every n'th: N/A enabled band

(Cancel)   (New list)

Band List Options *dialog.*

Band Scrolling

Every multiband image is loaded into HyperCube so that you can easily scroll through the bands in a movie like manner. The band scroll bar (buttons in the Windows version) is at the top left of the image cube window and works like any other Mac scroll bar. The current band number and corresponding wavelength are annotated just above the scroll bar as the cube is scrolled. If a specific band's status is disabled then its annotation is shown in red else it is black. A specific <u>band</u> can be given its own display window by scrolling the cube to that band and then choosing Duplicate from the Edit menu. It becomes completely seperate from the cube. Any band, whether it was loaded with the cube or not, may be loaded by double-clicking a band number in the Band List (see section Band List).

Band scroll bar, number and wavelength.

All controls except for Band Scrolling operate in the same manner in both versions.

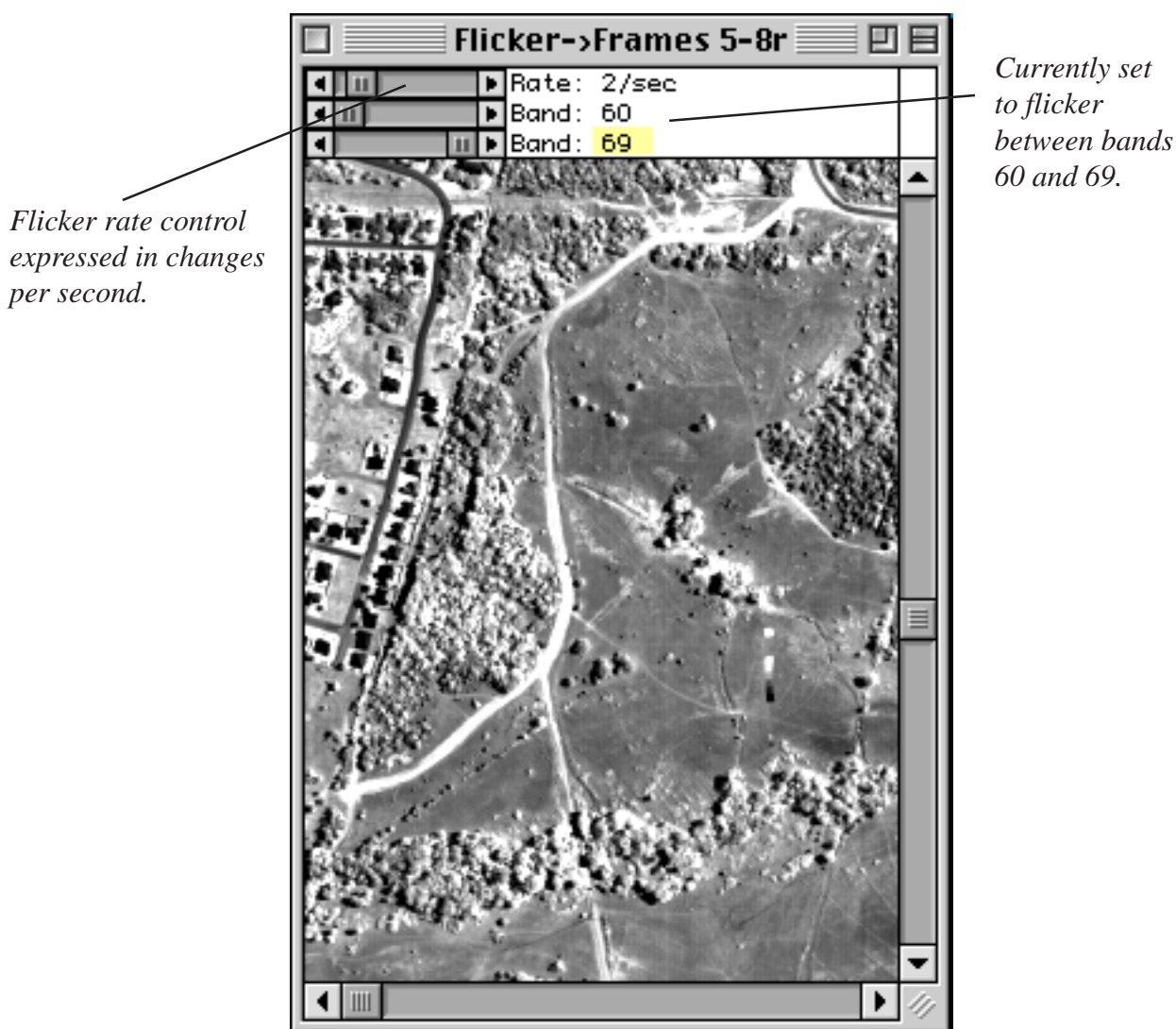<u>Windows</u> *version.*   <u>Mac</u> *version.*

Flicker/Superimpose

While band scrolling permits traversing an image cube in a contiguous fashion sometimes it's desirable to jump from one band to another to compare differences. HyperCube contains two functions that facilitate this. The first is the Functions->Flicker Cube menu and is enabled whenever a multiband cube window is front most and the second is Flicker/Superimpose which is enabled whenever at least one image window is open. Flicker Cube differs from Flicker/Superimpose in that it is specifically for the bands of an image cube and is adjustable, whereas Flicker/Superimpose is fixed to the two specific images (of any type) which drive it (see Flicker Image dialog below).
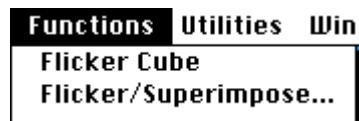
Flicker Cube creates a new window similar to the cube window but with three scroll bars at the top which control the rate and the two bands to alternate. Flicker Cube is an operator like graymap and if the cube window from which it is derived is closed then flicker disappears. The current flicker image can be printed and also copied.

The figure below is the flicker window generated from the previous band scrolling example. The flicker rate is nominal and some Macintosh models can not support the higher rates. Alternation continues even if the flicker window is not the front most window.

Currently set to flicker between bands 60 and 69.

Flicker rate control expressed in changes per second.

The second flicker type function, Flicker/Superimpose, is more general in that it permits any two image windows (and selection rectangles within) to be alternated between or superimposed one on another as if viewing through a porthole. The images are selected from two popup menus that appear when Flicker/Superimpose... is selected from the Functions main menu. The images do not have to be the same size, the viewing window is sized to their minimum. The image depth, i.e. 8, 16 or 32 bits, also doesn't matter. If one of the source images is modified such as by gray mapping or filtering then the change will be automatically reflected in the view window. Similar to Flicker Cube, if either source image is closed or rotated 90 degrees the view window will disappear. The window title alternates between the source images when Flicker is chosen.

Below is the dialog in which you choose the two images and selects: Flicker or Superimpose. If Flicker then the rate of alternation for flicker can be set. If the resultant flicker is too fast or too slow close the flicker window and generate another one. Superimpose places a rectangular window (or inset) of the first image upon a base consisting of the second image. This rectangle can be repositioned by clicking the mouse within it and dragging it anywhere within the window dynamically superimposing image 1 upon image 2. To reverse the ordering, reselect Flicker/Superimpose and swap Image 1 and Image 2 names in the popup menus. Superimpose is most useful when comparing an image band with a classification output image (see Products).

Flicker *alternates between the two images whereas* Superimpose *displays a rectangular viewing port of image 1 upon image 2.*
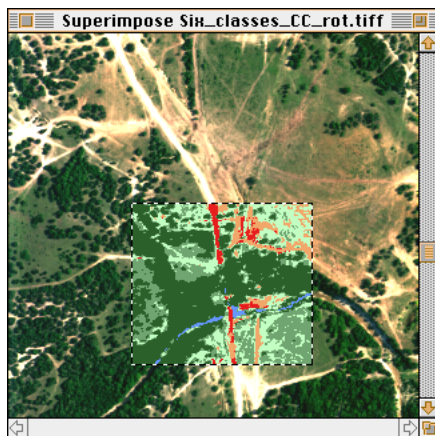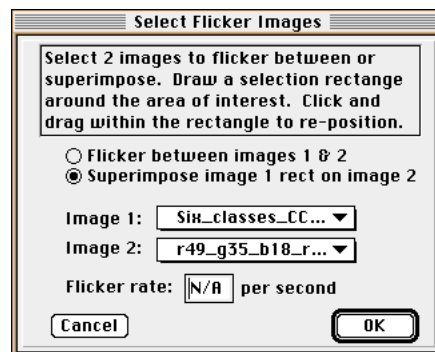

Image 1, Six_classes*..., has been selected to superimpose itself upon image 2, r49_g55_b18... The resultant window is shown at left in which the selection rectangle can be dynamically moved about the window.*

Constructing a Cube

HyperCube can generate a new image cube from existing images of the same spatial size having various pixel types (e.g., byte, short, long, float for mono and byte for color). Any image is a candidate including bands of an existing image cube. However, the new cube must be all of the same kind, either mono or color. Construction involves two steps: selecting the first band which determines the spatial size and adding subsequent bands to the cube. As new bands are added the pixel type of the cube automatically changes to accommodate the maximum number of bits/pixel required. Any candidate images should have been loaded with the Retain Source checkbox selected or else you will just be getting the 8 bit display values being added. HyperCube can also directly convert a true color image into a 3 band cube (see second method below). After the cube is complete you can save it via menu File->Save As... .
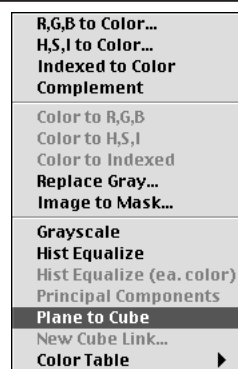
Here is an example of the steps required to produce an N band image cube from existing individual images:

1. Click on the first image component. This will become the first band.
2. Choose Image->Convert Image->Plane to Cube submenu.
3. Click on the second component.
4. Choose Add to Cube.. in Image menu. This is now the second band.
5. Repeat steps 3 & 4 for each additional band.
6. Select Windows->Show Band List menu and manually set up any wavelengths for each band (the default will be band numbers).
7. Choose menu File->Save As... and select Multiband to save the cube.

When converting a <u>color</u> image, an additional dialog is presented asking whether to directly generate a 3 band cube or the first band of a synthethic color cube as shown in the following dialog.
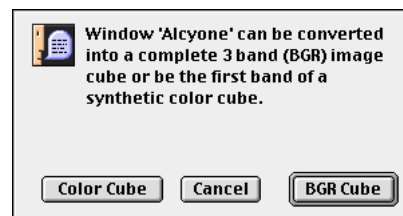
New bands may also be added to existing image cubes and if there is more than one cube window to which an image can be added (same spatial size) then the following dialog will appear after choosing Add to Cube... . The popup menu contains a list of same size cubes to which the image can be appended. Select the desired cube and click OK.
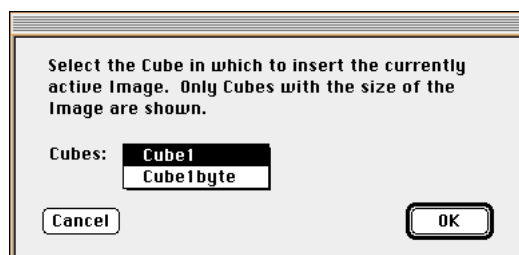
The dialog produced when Add to Cube has more than one possible destinations.

Color Composite

HyperCube provides a direct function to construct a true color image from a specific cube. The Image -> Cube Color Composite menu item allows the user to automatically generate a true color image for the currently active (front) image cube window and to link it to the cube for point selection and classification (see sections Spectral Plots and Classify Functions) breaking any previously linked color image. Two methods are provided via a submenu: Specific Wavelengths and Interval Wavelengths. The former finds the three closest bands to the wavelengths set in the Definitions... menu (see below) and the later integrates over three spectral intervals to generate the color components. In each case the data is extracted from the image file. Disabled bands (see Band List) are ignored so it is possible to have a null interval and HyperCube will warn you if this occurs. The resultant color image is correctly mapped so as to retain the original brightness relationships, i.e. the red, green and blue are normalized as a group.

The initial wavelength values for the Specific Wavelengths option red, green and blue component bands are 646, 545 and 460 nm (or µm) respectively. These correspond to a set of Kodak® color separation filters and may be changed by selecting Image->Cube Color Composite->Definitions. This produces the user dialog shown on the right. Interval Wavelengths options are similarly entered specifying the spectral range of integration to be used to form each color component. The Remap % determines the area about the median of the composite histogram that will be remapped to the [0,255] for each color. A good starting point is 98%. Use histogram equalization performs a nonlinear stretch of the components and is useful when the image cube has an asymmetrical histogram. Any edit changes made are global and affect subsequent cube color composites but revert back to those shown once HyperCube is exited.

If Edit->Options->Detailed Messages is checked then the composite histogram of the three bands is generated. This is similar to the histogram described in section Statistics except that it represents only the three bands or band intervals. A typical composite histogram is shown below.
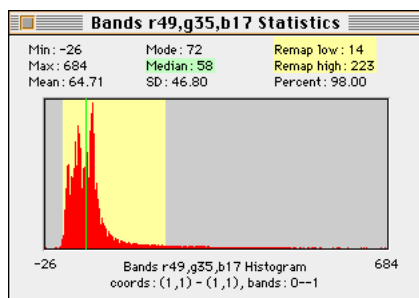
Bands 49, 35 and 17 were automatically selected from the active image cube file to form a composite color image. The histogram highlight represents 98% of the data in these 3 bands. File values between 14 and 223 will be remapped to the display range of [0,255] for each color.

Spectral Plots

Two methods of plotting spectral vectors from an image cube are provided: statically via Functions->Plots->Spectra(Selected Points) or Functions->Plots->Spectra(Dynamic) as a function of the cursor position. The former, described here, allows up to 64 points to be plotted along with library signatures superimposed upon the same plot (see Spectral Classifications section). The later allows the simultaneous display of the closest matching library signatures to the current cube vector (see Dynamic Displays and Plots in Spectral Classifications).
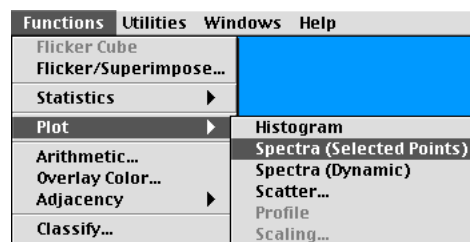
Points are selected by positioning the cursor and clicking the mouse while holding down the shift key. This draws a numbered colored cross hair on the cube face. The Plot window need not be open when points are selected and if no plot points exist when a Plot window is opened it will contain a message indicating what to do. If the wavelength file is incorrect no plotting is possible and the Plot window will contain a warning to that effect. Having the Info window open while you select points facilitates the process by dynamically showing the spatial coordinates. As points are selected they are graphed in the Plot window along with a color patch used for toggling them on and off. The number of color patches that can be toggled is determined by the size of the Plot window, resizing the window (lower right grow icon) displays more patches. All of the points can be erased at one time by choosing Clear from the Edit menu or selectively in the Classify dialog. When the cursor is positioned over a static Plot window the spectral and reflectance coordinates will be listed in an opened Info window.

The static Plot window contains several controls for customizing the plots: three check boxes, a standard button and dynamic colored push-buttons that toggle on and off their matching color plots. Cube bands that have been disabled in the wavelength file have a red highlight superimposed on them. Controls have the following definitions:

Scale: Rescale the vertical plot axis between the minimum and maximum of all of the vectors. Otherwise, the minimum and maximum will be set to the arithmetic range of the pixels.

[0,1]: Change the vertical axis labelling to the range [0,1] and map the plotted values to it. Also see Scaling... example below.

Hide Bad: If checked then suppress the plot wherever a band has been disabled. See section Band List.

Clear Lib: Remove any superimposed library signatures.

The dynamic Plot window is similar but does not have toggled points (see section Dynamic Displays in Spectral Classifications).
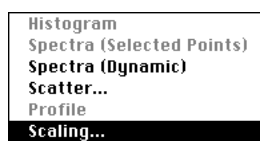
When a plot window is active Scaling *may selected from the* Plots *submenu to produce a dialog window which will result in* [0,1] *changing to* [0,1.5]*.*
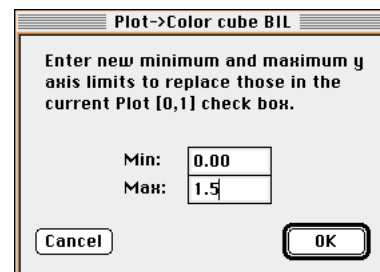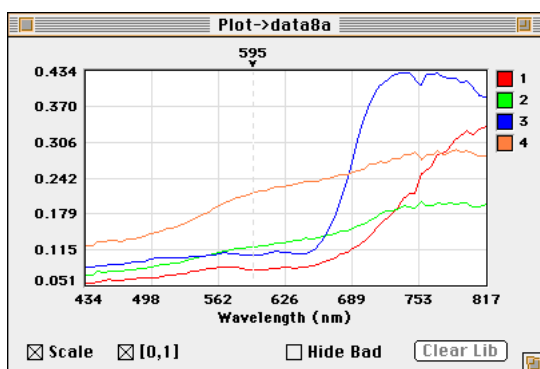
Four spectral plots from the 70 band image shown at the right. Each number corresponds to the numbered cross hair in the image. Several bands at 750 nm have been disabled in the wavelength file due to a sensor error. These points may also be selected from a linked red, green, blue composite image constructed from cube bands (see section Components to Color*).*
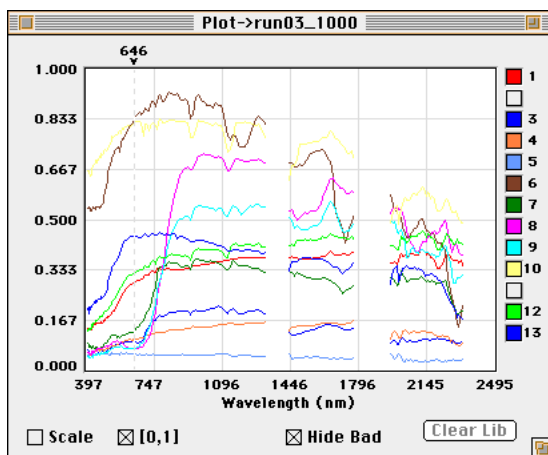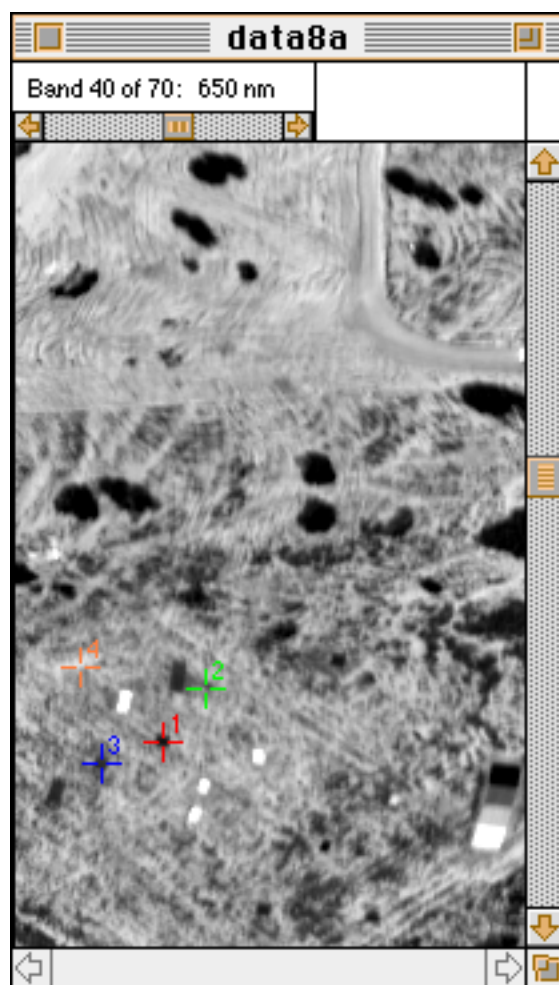
A different image cube spectral plot after resizing to show all of the corresponding color patches. Numbers 2 and 11 have been toggled off. Because Hide Bad *is checked, plots within the two major water absorption regions are not drawn (disabled in the* Band List*). The correspondence between a point number and its color may be changed using the* Color Selector *item in the* Options *menu.*

Changing Selection Point Colors

When HypeCube is launched the correspondence between a selection point number and its color is set from an internal default table that will be used when plotting spectral signatures and classifying (see Products in Spectral Classifications section). The color sequence is arbitrarily repeated in groups of 10 with each group slightly different in value. Therefore, although points 1, 11, 21, 31, 41, 51 and 61... appear as the same red, they are not. The point # to color correspondence can be customized at any time by choosing menu Edit->Options->Color Selector. Up to 1024 colors may be uniquely redefined. When a change is made all windows using selection point colors will be updated to reflect the change. Classify output products that have already been generated will not be changed (Overlays and Masks <u>will</u> use the new color on a subsequent classify). However, products can be reassigned colors and labels after the fact by using menu Utilities->Color to Name Editor (see Dynamic Displays and Plots). A color that has been copied to the clipboard can be pasted as the selected color (see section Copy, Paste & Duplicate). Also, see Classify->Options->More Options->Use source spectra color in section More Options as an alternate method of specifying class map colors.
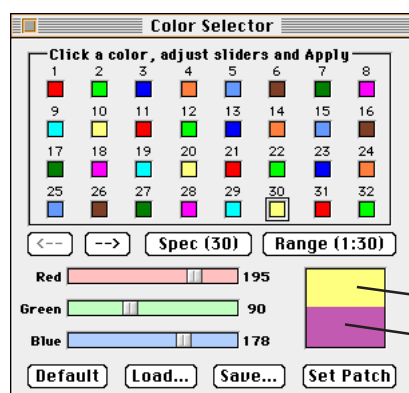
Color patch #30 has been selected and the sliders adjusted to a new color. The components of each color are shown to the right of the sliders and range from 0 to 255. You can not select a gray value (i.e. all three colors with the same value). The Selector *will automatically modify the blue value to be different.*

Current palette color.

New palette color.

To change a single color click on its patch to highlight it (a small frame is drawn around it) and adjust the three sliders for the desired color. The top half of the palette shows the current patch color while the bottom half of the palette changes to show the new color.

|	
<--, -->	These buttons are used to navigate through the 1024 colors moving fowards and backwards 32 colors at a time.
Spec (N)	Generate a <u>rainbow</u> spectrum in the first N colors ranging from red through yellow, green, cyan and blue. N is the currently toggled color patch. If no patch is selected then N will equal 1024.
Range (1:N)	Generate a smooth range of colors between the colors in patches 1 and N. If no patch is toggled then 1024 colors will be generated.
Set Patch	Change the patch color to the new palette color and update all windows.
Default	Change the selected color patch to its original HyperCube launch color and update all windows. If no patch is currently selected, a dialog appears asking you if all patches are to be set to their defaults.
Save/Load...	The current color key correspondence is saved/retrieved to/from an ASCII file. The output is the same format used in the Color to Name Editor, i.e., red, green and blue values followed by a name which in this case is the index number. The default file extension is "c2n".

Spectral Libraries

Format

The HyperCube program currently recognizes a spectral signature format developed at TEC for the recording of field and laboratory spectra as well as a simple list of wavelength and value pairs. One signature represents the spectra of a single material. A set of signature files constitutes a library of materials. The various signatures belonging to a library do not have to be related in any manner, they are often grouped together to correspond to a particular sensor run.

The format of a TEC signature file is very straight forward: a small header followed by an ordered set of linearly spaced reflectances. All of the elements are in ASCII text and can be easily edited in a word processor or spreadsheet application. Almost any ASCII text file can be read, edited and resaved in HyperCube, including signature files (see section Editing Text Files under Miscellaneous).

A library is a text file whose elements are signature file names, i.e. a library contains no data except file names. The library file and all of the signature files must be in the same folder. A large (several hundred) library file can be initially generated by using the Finder to select and copy a group of signature file names from View by Name and then pasting the names into a document window of a word processor. There is a Finder limit of the number that can be selected at one time and this may need to be repeated.

The relationship between the library elements is depicted below:

Each spectral signature, regardless of size, uses the following format:
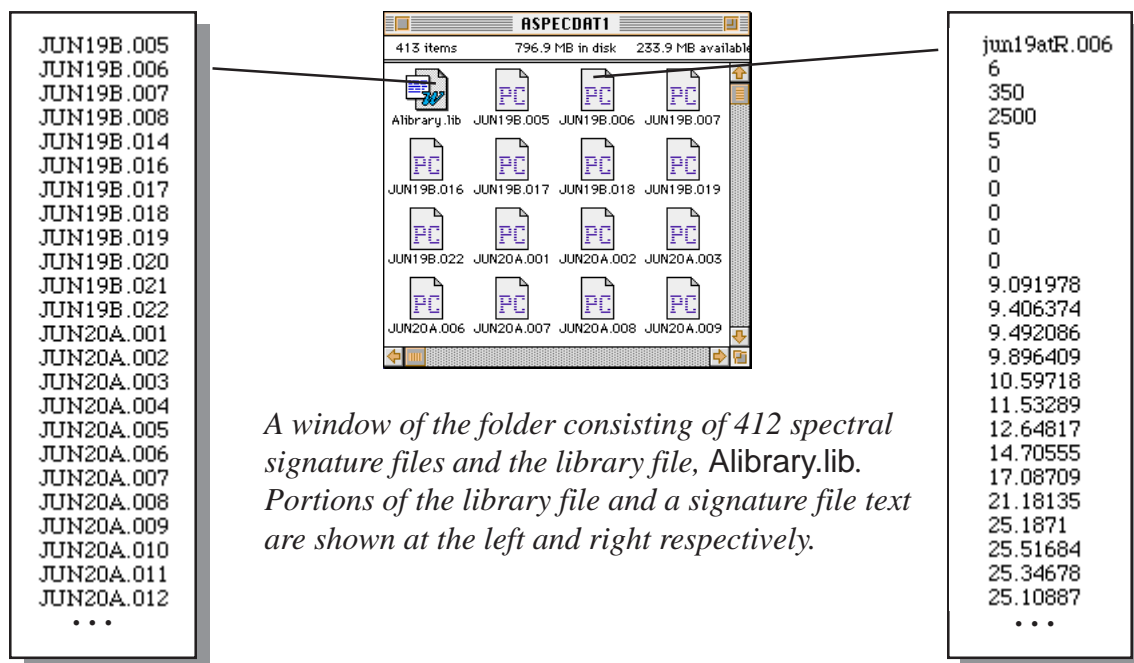
A window of the folder consisting of 412 spectral signature files and the library file, Alibrary.lib. *Portions of the library file and a signature file text are shown at the left and right respectively.*

Alibrary.lib *file with an entry for each signature file name.*

This spectral signature file contains a total of 431 = (2500-350)/5+1 reflectance values.

Line:	Parameter:	Example:
1	File name	jun19atR.018
2	Spectrum number	18
3	Starting wavelength (nm)	350
4	Ending wavelength (nm)	2500
5	Wavelength increment (nm)	5
6	Scan index number (hr)	0
7	True solar time (hr)	0
8	Solar azimuth (degrees)	0
9	Solar azimuth (degrees)	0
10	-- not used --	0
11	First wavelength reflectance (%)	11.15
12	Second wavelength reflectance (%)	11.05
.	.	
N	N-10th wavelength reflectance	##.##
.	.	
438	428th wavelength reflectance	25.27
439	429th wavelength reflectance	26.61
440	430th wavelength reflectance	0
441	431th wavelength reflectance	0

Lines 3,4 and 5 implicitly determine the number of reflectances.

These last two measurements are filler in this example.

For TEC format spectra HyperCube only needs the information in lines 3,4 and 5 plus the reflectance values.

An alternate format of a spectral signature is simply a list of wavelength and value pairs. The wavelengths and values can be in any units as long as the wavelengths are increasing. Each pair should be space or tab separated followed by a carriage return. HypeCube determines the format of a spectral signature by first trying to parse it as if it were in the TEC format. If that fails it assumes it is wavelength, value pairs. An example of the later is shown below.

Wavelengths   *Values*

```
399.000000  0.028000
402.000000  0.027000
405.000000  0.018000
408.000000  0.023000
412.000000  0.021000
415.000000  0.024000
419.000000  0.028000
422.000000  0.028000
426.000000  0.027000
429.000000  0.029000
433.000000  0.033000
437.000000  0.035000
440.000000  0.034000
444.000000  0.035000
448.000000  0.036000
452.000000  0.039000
```

Plotting Signatures

Viewing a spectral signature requires first opening a library by choosing Open As..., (Library) from the File menu. This brings up the standard Mac file dialog showing only text type files. Select the library file containing the signature names (Alibrary.lib in the examples so far) and click Open. A new sizeable window containing a scrollable list of the signatures within the library will appear. Click on any signature name, select Plot Selection and a subsequent window, also resizeable, will open showing a list of reflectances with the corresponding graphical plot. The average of two or more signatures can be plotted by double-clicking the desired signatures, setting Average selections and then the Plot Selections button. Double-clicking an item toggles its checkmark on and off.

Double-clicking an item toggles it on so that an average of all items with a checkmark can be plotted.
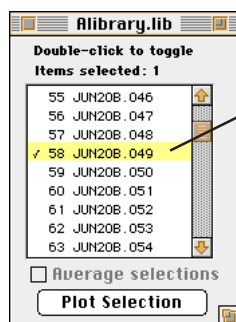
Scroll to each signature of interest, click on it to highlight it, and then click the Plot *button.*
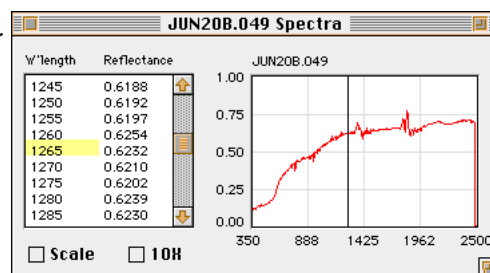
The corresponding signature to the library list after clicking in the graph at the 1265 nm position.

Spectra plots have two check boxes that affect the appearance of the graph. The Scale option toggles the vertical axis (reflection) of the plot range. If unchecked the range is fixed to [0,1] else it is set to [min,max] of the reflectances of the viewable graph. The second check box, 10X, expands the horizontal axis 10 times centered upon the current spectral marker, if any. Clicking within the graph area sets the marker and forces the list to scroll to the corresponding wavelength. Conversely, clicking the list sets the marker within the graph. Spectral plots may be printed by choosing Print... from the File menu and a snapshot of the window can also be copied to the clipboard. The list contents can be written to a file in the form of a library (see section Spectra Libraries) by selecting Save from the File menu or copied to the clipboard via Edit->Copy.
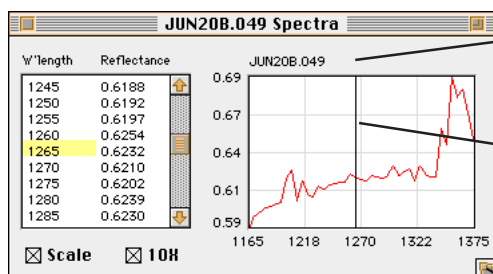
Line 1 of the signature file.

Click within the graph to set the wavelength marker; the matching list element will be highlighted.

The previous spectra plot example repeated with both check boxes set.

Click and drag icon to resize.

Superimposing Signatures

    If HyperCube is running in system 7.5 or higher then the Mac's "drag and drop" feature can be used to overlay signatures on a common spectral plot. First open a multiband image and choose Functions->Plot->Spectra (Selected Points). Then click the mouse in a signature plot and while holding down the mouse drag the rectangular outline, which will appear, over the spectral plot graph and release the mouse. The signature will then be plotted in black superimposed with any selected cube points and will have its own toggle button. Up to 8 signatures can be "dropped" on the spectral plot window. The signature will be re-sampled to match the range of the multiband image.
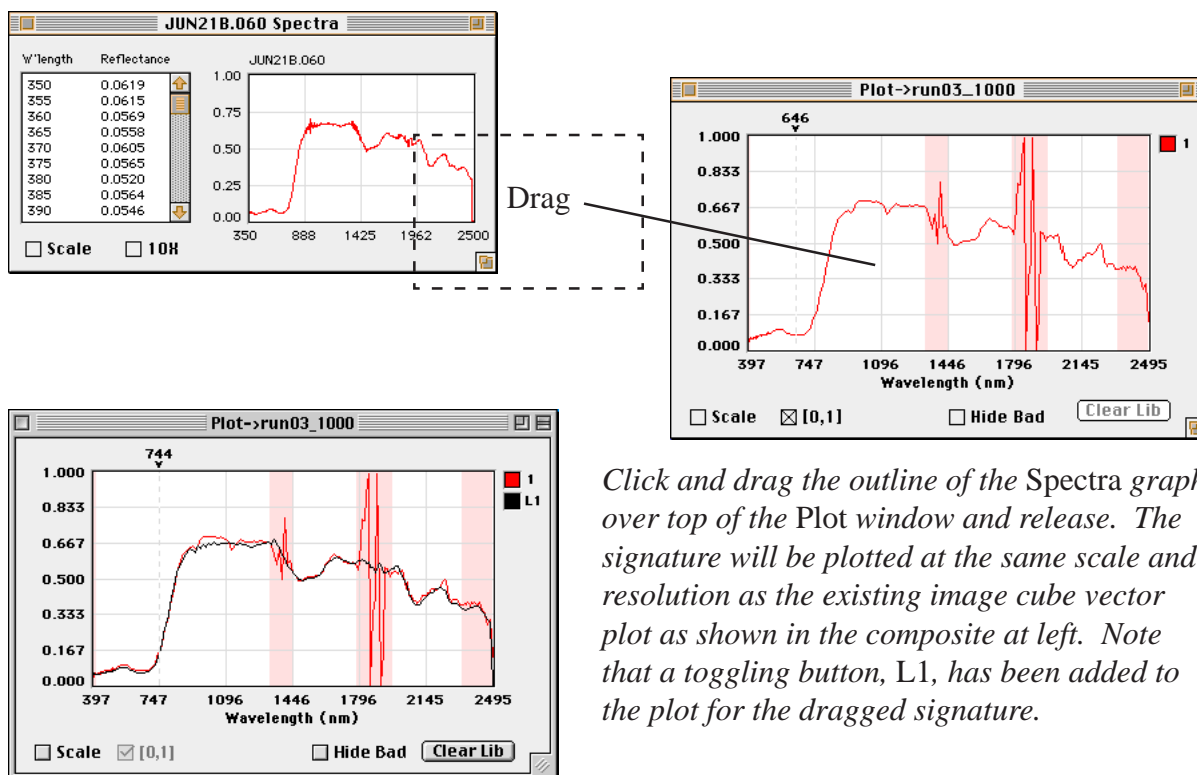
Drag

Click and drag the outline of the Spectra *graph over top of the* Plot *window and release. The signature will be plotted at the same scale and resolution as the existing image cube vector plot as shown in the composite at left. Note that a toggling button,* L1*, has been added to the plot for the dragged signature.*

Spectral Classifications

Algorithms

Spectral classification is the process of segmenting a multi or hyperspectral image into regions of similar spectral statistical characteristics or of finding the best match between pixels given a specific metric. HyperCube is designed to perform both. For example, Mahalanobis classify uses previously extracted statistical classes and decides which class, if any, a pixel belongs to, whereas Vector angle directly compares image pixel vectors with selected pixel vectors. The criteria of a match is the closeness of the comparison. This is a function of the algorithm used to define the closeness and HyperCube provides several from which to choose. No one algorithm is always better than the others. Some are very sensitive to noise, e.g., Mahalanobis distance, derivative difference, and others, e.g., Euclidean distance, are not. The best technique is to experiment.

Listed below are the algorithms from menu Functions -> Classify along with their arithmetic definitions. Summation extends over all <u>enabled</u> bands (see Band List).

Abs diff	Σ abs(reference - signature).
Corr coef	Covariance(reference,signature)/(sd(reference)*sd(signature)).
Deriv diff	Σ abs(delta(reference) - delta(signature)).
Euclid distance	Sqrt(Σ (reference - signature)^2).
Fourier	Compare corresponding harmonic terms. See Options.
Fourier (no DC)	Same as Fourier excluding the zero (average) harmonic.
Fourier phase	Compare only the phase of corresponding harmonics.
Fourier phase delta	Compare the phase differences from harmonic to harmonic.
Kmeans	Unsupervised classifier that segments an image into a user specified number of classes.
Mahalanobis classify	<u>Statistical</u> difference between each image vector and the mean vectors of a set of covariance matrices.
Mahalanobis distance	Statistical distance beween reference points and image points.
Mahalanobis list	A listing of the statistical difference between reference points and the mean of a set of covariance matrices.
Matched filter abundance	Generate a gray shade output image consisting of projections of the source data against a constructed matched filter of a signature(s).
Matched filter mask	Generate a mask over the image cube wherever the abundance exceeds a set tolerance (see description on next page).

Template   Max(abs(reference - signature)). Useful for finding very
                  similar signatures.

Vector angle   N-dimensional angle between reference and signature.

 Closeness is defined differently for each of the algorithms. Euclidean distance uses minimum as the criteria whereas the correlation coefficient approaches 1 for the best match. This is automatically taken into account by HyperCube but it is not possible to equate the numeric values of one method with corresponding ones in another method because the distribution of each method is totally different.

 It is important to remember that a classification extends only upon bands whose status is Enabled in the Windows -> Show Band List and is spatially confined to any selection rectangle. The Band List makes it easy to test the contribution of a given band by running a classification, enabling/disabling the band, and rerunning the classification to see the effect. Any statistics required by the algorithm will automatically be recomputed.

Specific Algorithms

 <u>Matched filter abundance</u> does not use a tolerance, instead it directly generates a data image where each pixel has a value approximately in the range ~[-2,2]. It is possible to have numbers greater than 1.0. If the pixel used as the matched filter signature is also part of the classification region then it will have a value of exactly 1 at its position within the output image. The matched filter and abundance are defined as:

$$\text{matched filter} = (\text{cov})^{-1}(\text{signature-m})/((\text{signature-m})^{T}(\text{cov})^{-1}(\text{signature-m}))$$

abundance = inner product of each data vector minus m and the matched filter.
where:
  cov is the NxN covariance matrix of N bands of imagery
  m is the Nx1 mean data vector
  signature is the Nx1 pixel vector, N being the number of enabled bands.

When Matched filter is chosen as the classify Method the program must first compute the band to band covariance of the enabled file bands of the image cube or the corresponding subimage selection rectangle, if any. The determination of the statistics region is via Classify -> Options -> Stats Region (see Mahalanobis description, next). The computation is done once and the covariance matrix remains the same even if the current selection rectangle is moved or a different selection rectangle is drawn. This allows the statistics for the covariance to be generated for an image region other than the area chosen for the matched filter classification region. To force a recomputation of the covariance choose Classify -> Options -> Redo stats. Also, changing the state of enabled/disabled bands in menu Windows -> Show Band List will cause a recomputation of the statistics on the next classify.

Mahalanobis distance is a generalization of Euclidean distance that uses the statistics of the imagery. Specifically, the Mahalanobis distance squared between 2 vectors, X and Y is defined as:

$$D = (X - Y)^T (cov)^{-1} (X - Y)$$

where:

D is the resulting scalar (i.e., 1xN times NxN times Nx1 matrices)
X and Y are each pixel vectors of length N
$(cov)^{-1}$ is the inverse covariance matrix of N bands of imagery
and N is the number of enabled bands.

In HyperCube, X and Y are defined by the various Mahalanobis classify methods as:

Mahalanobis classify - X is each image pixel vector.
Y is the mean pixel vector of the region used to generate a covariance matrix. There can be one or more regions, each producing a mean vector and covariance matrix. Classification involves computing the Mahalanobis value of an image vector for each region and choosing the region (class) with the smallest value within a user supplied tolerance. Selection points are not used. Regions are defined by user selection of a mask image in Classify -> Options -> Stats Region. The statistics (covariance) are automatically generated by HyperCube based upon the mask definition.
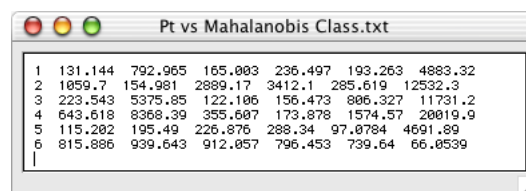
Mahalanobis dist - X is each image pixel vector.
Y is one or more selection points.
The image statistics are the same as defined for Mahalanobis Classify except that the mean vector is not used and only one statistical region is used. This produces the statistical distance between one or more selection points and an image, not the distance between the statistical class mean and image.

Mahalanobis list - Same mathematics as Mahalanobis Classify except selection points are used and the output is a numerical listing showing each selection point's distance from the mean vector for each of one or more classes.

Six selection points and their Mahalanobis
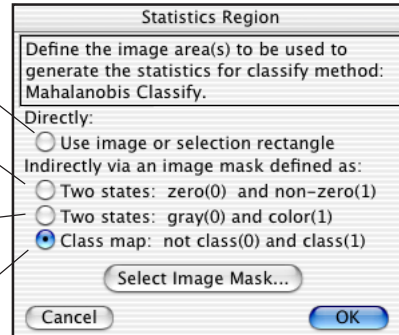*value for each of six statistical regions.*

The image area used for computing the covariance matrix depends upon Classify -> Options -> Stats Region. This displays the dialog shown below. There are 4 ways to

Use every pixel within the image or selection rectangle.

Skip image pixels wherever the mask has a zero, use image pixels wherever mask is not zero. Produces a single covariance statistics class.

Skip image pixels wherever the mask is gray (including zero) and use image pixels wherever mask is color. Also produces a single covariance statistics class.

Mask must be a class map (section Image Products*).*
Each color class produces a covariance statistical class.

Statistics Region

Define the image area(s) to be used to generate the statistics for classify method: Mahalanobis Classify.

Directly:
◯ Use image or selection rectangle
Indirectly via an image mask defined as:
◯ Two states: zero(0) and non-zero(1)
◯ Two states: gray(0) and color(1)
◉ Class map: not class(0) and class(1)

Select Image Mask...

Cancel   OK

define the regions. The first type computes the covariance statistics using all of the pixels within the image cube or a portion of the image if a selection rectangle is present. The remaining three methods are also based upon the entire cube or a portion but do not necessarily use every pixel. Instead, the pixels used for generating the covariance statistics are determined by an intermediate image mask that spatially matches the image cube (or current selection rectangle). A simple way to generate a class map image mask for the fourth method so the currently displayed sequence band face is used as the selection mask (additional information below).

Statistics are automatically computed when the Classify button is executed (section Classify Function) if there has been any change in the Classify -> Options -> Stats Region -> Select Image Mask selection. They remain in effect so that a different area on the image cube can be used for classification based upon the same statistics. Also, any enable/disable change in a band list (see section Band List) will cause a re-computation of the statistics. Button Classify -> Options -> Redo stats forces a re-computation and is useful if the selected image mask was generated as a class map sequence (see section Classify Options). Then the current sequence class map is taken as the sequence cube face and can be used as follows: Select a new face, Redo stats, and then Classify to see the effect of a mask change within the sequence (saves having to repeatedly go to Classify -> Options -> Stats Region -> Select Image Mask each time the sequence face is changed).

Some additional considerations: the convariance matrix is very sensitive to noise. Noise, being defined here, is anything that is not representative of that class. Even a single outlier, such as striping in the image, will skew the statistics and produce a non representative covariance matrix. If the imagery is noisy then try using fewer widely spaced bands (see section Band List). If a two state mask image is selected or in use and Mahalanobis Classify or List is chosen as the Classify method then the outputs reduce to one statistical class.

<u>Kmeans</u> is an unsupervised classifier that does not use any selection points and requires minimal input. The algorithm segments the image into n classes by initially guessing the mean vectors of the n classes. It then scans the image and determines (using one of the three classification metrics listed below) which class each image vector belongs to and a corresponding set of new mean vectors. This procedure is repeated until the number of image vectors that changed classes is below a certain minimum or a maximum number of iterations has occured. The number of classes and both termination criteria as well as other parameters are user selectable. Choosing Options in the Classify dialog while the classify Method is Kmeans displays the dialog shown below (note: the Options dialog associated with Kmeans is different from the options shown for the other classify algorithms).

The number of desired classes is limited to 500 or less. The actual number of classes generated may be less than the number requested. Use menu Functions -> Plot -> Histogram *on the classify output to list the classes.*

A Progress *window dynamically displays the iteration count and the number of class changes. When the number of iterations reaches this value* Kmeans *classify stops and outputs the results up to this point.*

When the number of class changes is less than or equal to this value classify terminates.

K-means Options

K-means estimates the class mean vectors and then repeatedly updates the means until max iterations or convergence.

Enter:
 Number of classes: 16
 Maximum # of iterations: 50
 Convergence # of changes: 0

☑ Collect iterations into a sequence
☑ Plot final class mean spectra
☐ Save final class mean spectra as a library
Classification metric:
 ○ Correlation coefficient
 ◉ Euclidean distance
 ○ Vector angle

Use final class means as class colors:
 ☐ Natural
 ☑ Saturated

Cancel   OK

Remaining options:

Collect iterations: Each iteration (classify) is assembled into a class cube that can be scrolled to view the convergence sequence.

Plot final class...   Each mean vector forms a spectral plot window.

Save final class...   The final class mean vectors are saved as a spectral library.

Classification metric   The algorithm that is used to determine which class an image vector belongs to by comparing it the mean of each class.

Use final class means as class colors:
   Natural - Determine mean class colors directly from mean vectors.
                        Saturated - As above, but saturate the color.
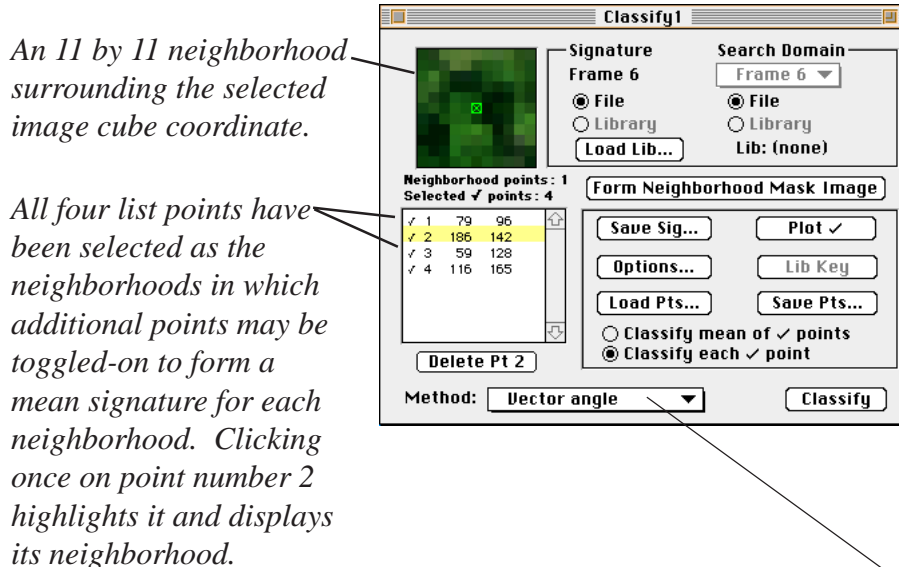   Yields an n class true color image.

Classify Function

The HyperCube vehicle for spectral classifications is initiated by choosing the Functions->Classify menu and is available whenever a cube or a linked color band composite window is front. Each image cube can have at most one Classify window. A classification scenario has three basic steps: decide on the signature to be matched (what to look for), choose the search domain (where to look), and pick the method (how to look). The output product is determined by the domains and numerous options. The simplest classification is to shift-click on a point in the image cube window, double-click on it in the Classify list, choose one or more neighborhood points and generate an overlay image of the current Search Domain cube face band using Vector Angle as the Method. This is the default setting of the Classify window and does not involve a library. A Classify window is logically connected to its image cube and closes when the cube window closes.
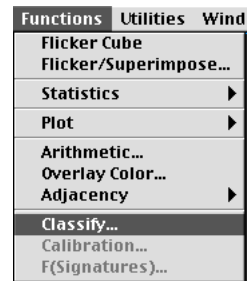
As you select cube points their coordinates are added to the Classify list. These points are then used to reference one or more surrounding neighborhoods from which you can toggle one or more signatures to be averaged together. The list coordinate is the central pixel in the neighborhood and has the same color as the cube selection point reticle but with a transparent interior. It is also the same point that is graphed if a plot window is open (see Spectral Plots). By clicking the mouse within the neighborhood, points may be added or removed from the signature mean. Shift-clicking adds or removes a rectangular region between the shift-click and last <u>non</u> shift-click point. Any combination of neighborhood points is allowed. If more than one pixel is toggled-on then their mean is formed. Multiple <u>neighborhoods</u> are selected by double-clicking each list point number. Clicking the Plot Mean button produces a graph of the resulting signature and this graph may be superimposed upon an open spectral Plot window (see Superimposing Signatures).

The neighborhood image is generated from the band currently shown on the Signature cube window and therefore will change as the cube is band scrolled. However, if a linked color image has been constructed from cube bands (see sections Components to Color and Color Composite) then the neighborhood is extracted from the linked color composite.

An 11 by 11 neighborhood surrounding the selected image cube coordinate.

All four list points have been selected as the neighborhoods in which additional points may be toggled-on to form a mean signature for each neighborhood. Clicking once on point number 2 highlights it and displays its neighborhood.

The Search Domain does not have to be in the same cube as the Signature which is fixed to the selection point cube.
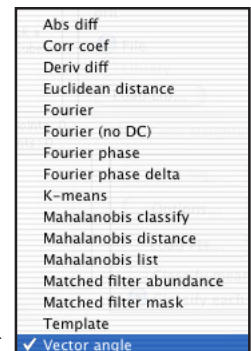
The following is a description of each Classify window control.

Signature: The signature source is always taken from the point selection cube and
                  its name is shown beneath Signature. The only exception is when
                  Library is chosen (see below).
    File   Use a selected signature signature from the image cube.
    Library   Use a selected signature from the current library as the signature.

Search Domain: The popup menu beneath this title is used to select the cube in which
                  to search. The Search Domain does not have to have the same
                  spatial or spectral dimensions as the Signature source. A different
                  cube is only permitted when Signature and Search Domain are
                  both set to File. Also see Library below for another exception.
    File   Match the selected signature against image cube signatures.
    Library   Match the selected signature against all signatures in the library.

Method: Popup menu of available search algorithms.

Library...   Open a standard file dialog to select the library to use. The name
                  of the library will replace Lib:(none) in the Classify window.

Lib Key   Active after a library load. Generates a window showing the
                  correspondence between a toggled-on library item and the
                  color that will be used in the classification. See the later section on
                  Products and earlier section Changing Selection Point Colors.

Form Neighborhood Mask Image:
                  A separate gray image is generated spatially matching the current
                  selection area and having 255 gray values at each toggled selection
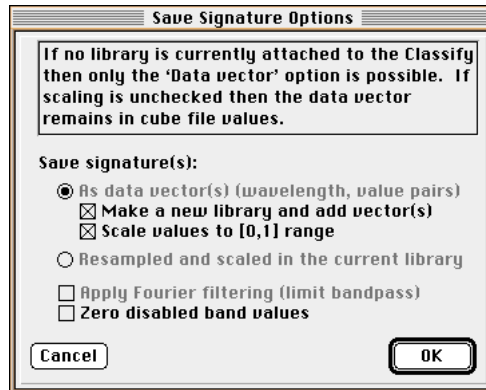   point and neighborhood with zero elsewhere. See ROC Curves.

Plot √   Generate a spectral signature plot of toggled-on neighborhood points.
                  If Classify each √ point is checked then the plot will contain each
                  neighborhood and their mean. If Classify mean of √ points is
                  checked then one mean of all the selected list points and their
                  neigborhood points is formed and plotted. If Method is Fourier then
                  the plot is pre-filtered.

Options...   Open the Classify Options window.

Delete Pt #   Delete the highlighted selection point from the list and move lower
                  points, if any, up. Also affects any current plots.

Load/Save Pts... Save the current list of selection points including their
                  neighborhood definitions. Upon loading all points are
                  automatically toggled on.

Save Sig...   If Signature is set to File, then the currently toggled on selections
               can be saved as a spectral signature of raw values, as a new library
               or added to the currently opened library. The following dialog is
               presented.

```
════════════ Save Signature Options ════════════
┌─────────────────────────────────────────────┐
│ If no library is currently attached to the Classify │
│ then only the 'Data vector' option is possible.  If │
│ scaling is unchecked then the data vector          │
│ remains in cube file values.                       │
└─────────────────────────────────────────────┘

Save signature(s):
  ◉ As data vector(s) (wavelength, value pairs)
     ☒ Make a new library and add vector(s)
     ☒ Scale values to [0,1] range
  ○ Resampled and scaled in the current library

  □ Apply Fourier filtering (limit bandpass)
  □ Zero disabled band values

[ Cancel ]                              [  OK  ]
```

Classify mean of √ points
Classify each √ point
               These form a pair of mutually exclusive radio buttons that determine how
               the toggled-on (√) points are used in the classification process. The default
               is Classify each. Then each √ point generates a classification region in
               competition with the others. That is, cube pixels are compared with each
               point and the closest point is defined as the proper classification. The
               program requires that the classification criteria be set to Threshold in the
               Classify Options dialog. If no point matches within the Threshold then
               that pixel remains unclassified (see the next sections, Options & Products).
               If Classify mean is chosen then all of the √ points are averaged together to
               form a single mean signature that will produce a singe classification region.
               Any classification criteria may be chosen in Classify Options.

               Note that if there is only one √ point then either of these choices will produce
               the same result if Threshold is the criteria. Also, any √ point that has
               multiple neighborhood points selected will still form an average of that
               neighborhood to represent the signature of that √ point regardless of the
               choice here.

               The color that is used to delineate classification regions is the same as each
               selection point color for Separate and the color of the first selection for
               Mean. The colors can be changed using the Color Selector described in
               section Changing Selection Point Colors.

Classify/ROC Curve:
               Perform either a classify or ROC curve. This button toggles between these
               two as a function of the ROC curve checkbox in Classify Options dialog.

Classify Options

Pressing the Options... button in the Classify window opens the following dialog window.

Each classify Method *has a theoretical mathematical range and is shown along with any parameter that must be set for that* Method.

There are 4 ways to compare a signature within the search domain. If Classify each *is set then the criteria will be forced to* Threshold.

The output format is determined by these radio buttons and the checkboxes.

The number of signatures to list when Search Domain *is a library.*

More Options... displays another dialog showing additional classification products.

Selected Method Parameters:

This section shows the currently selected classification Method and allows you to change a specific parameter associated with that method. Presently, only Deriv diff and the various Fourier methods have such a parameter and are described below. The Theoretical best and worst tolerances refer to the mathematical closest and furthest difference between the signature and search domain vectors. These are used to set thresholds and sequence limits and vary depending upon the alogorithm (Method). For instance, Vector Angle and Fourier Phase range from 0 to π radians whereas Corr Coef ranges from 1 (best) to -1 (worst). The various metric distances such as Abs Diff are normalized to the range [0,1] regardless of the arithmetic range of the search image cube. Euclidean distance is normalized to the sqrt of the number of enabled bands. The Methods which allow parameters are described below.

Deriv diff delta: The sample distance from signature point to point used to compute the derivative. Generally, signatures are quite noisy and using a delta greater than 1 provides a better indication of the true slope. However, if there are few or widely spaced bands some important signature variation may be skipped.

Fourier terms: The low order number of Fourier transform harmonics to be used in comparing signatures in the frequency domain. The greater the number of terms the finer the discrimination and sensitivity to noise. The smaller the number the greater the filtering and immunity to noise. A good starting point is 8 and this value must not exceed the number of enabled bands divided by 2.

Redo stats: Available with classify methods Mahalanobis and Matched Filter.
Stats Region: Used to define the region for generating needed covariance statistics. See description in section Algorithms.

Classification Match Criteria:

The four mutually exclusive radio buttons in this area specify how the mathematical difference between the source signature spectra (vector) and the search domain spectra is used. The first one, Threshold, is the most direct. It allows you to enter a tolerance in the range given by the Selected Method Parameters section just described. Then, each computation is compared with this tolerance and if it is between it and best then the image cube pixel is defined to be classified as the source spectra, otherwise not. If more than one point is being classified, i.e. Classify each, then the closest point to the best within the tolerance, if any, determines the classification.

Threshold: A value in the range: [best,worst] that serves as a breakpoint. Computations within [best,value] are classified, outside this range are not. Example: if the Method is Vector Angle then the tolerance range will be listed as 0 (best) and π (worst). Setting the tolerance to .08 means that all vector angle differences between 0 and .08 will be considered to be "classified". The color given to a classified region is the same as the selection point color and may be changed via the Color Selector. Threshold is the only criteria possible if Classify each is selected.

Percentage: Generate a color image using the selection point color to indicate the closest percentage of the entire distribution of the match. For example, if the percentage is 10% then the upper (best) 10% matching pixels are defined as classified.

Equal Distribution: Generate an N color output image where N is chosen by you. The entire match distribution relative to the source spectra is divided into N equal areas. Each area then becomes a classification region. Unlike the other criteria, the closest region has the color of patch 1 in the Color Selector, the second closest region has the patch 2 color and so on.

Countable: Generate an output image with the N closest matches delineated. For example: if N equals 50 then the 50 closest matches will define the classification region. Good for showing very small targets.

Classification Output(s):

This section defines the form and number of output products. There are 2 types and both produce an image. Each generates the appropriate color classification regions for the Match Criteria listed above. Their difference is how an unclassified region appears. The Color Mask leaves these areas black while the Color Image shows the current cube face band where there is no classification. Both selections produce a true color image. Note that the colored regions correspond exactly to the color currently defined for points in the Color Selector. If the colors don't seem appropriate for the given classification change them in the Color Selector and redo the classification. Also, see the color note above regarding Equal Distribution. Flicker/Superimpose is very effective in dynamically comparing a classification output with a source image (i.e. the linked Color Composite or an image cube band).

Color Mask: The output image has a color where a classification is define as successful and black otherwise. The only output possible if Equal Distribution is selected.

Color Image: A copy of the current cube face band will be used as a base image and the classified regions will be overlaid.

Generate Sequence: If checked then the 2 tolerance values entered below the checkbox define a range of Match Criteria in which a sequence of classification images will be generated. The number of outputs is user selectable. The outputs are arranged into a synthetic cube for display (see Products). The example below will generate a synthetic cube of 10 images where image 1 represents a classification tolerance of 0 and image 10 a tolerance of .5 (in this case radians).

ROC curve: This option is available if Classify mean of √ points and Generate a sequence are both selected. No image output is

```
┌─Classification Output(s)──────────────────┐
│ ○ Color mask (black where no match)        │
│ ◉ Color image (cube face shows where no match) │
│ ⊠ Generate a sequence, Enter # of outputs: [10] │
│    Starting tolerance:  [0.000]    ☐ ROC curve │
│    Ending tolerance:    [.5]      [ ROC Mask... ] │
└────────────────────────────────────────────┘
```

generated, instead a plot of the number of classify hits versus tolerance is generated. See ROC Curves section for details.

ROC Mask... Displays a popup menu dialog allowing you to select the "ground truth" mask image to be used in generating ROC curves.

Library Parameters:
This last section is applicable only when the Search Domain is Library. In this case the output is not an image but ordered sets of matching library signatures.

Closest lib matches: You can specify how many library signatures to list (see the Products section).

Lib:File match lists: Generates a list of the library elements found in the image and a list of the source library elements that didn't match.

Ignore Disabled bands: Available when Signature and Search Domain are both set to Library. In this case you may want to perform matching at every spectral band value by including any wavelengths that are disabled in the Band List.

Enable Dynamic Display: Available when Signature is File and Search is Lib. As the cusor is passed over the Cube or Link image a real time display of the closest library matches is dynamically displayed. If a dynamic spectral plot is enabled, then the closest 2 library spectra will be plotted.

When performing a classification you may request several output products that can be generated in addition to the class map. By selecting More Options... in the Classify Options dialog the following dialog appears.
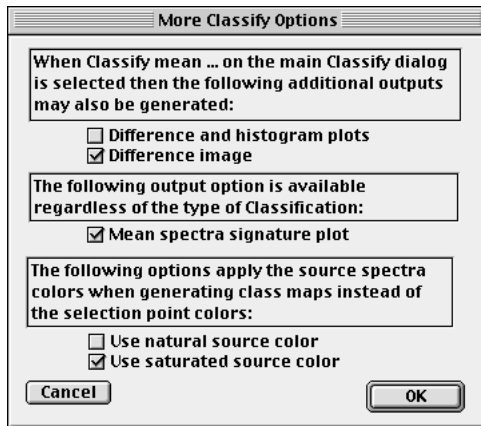
Three options have been selected: produce a difference image between the selected classification point and each spectral signature within the image cube, compute the average of all spectral signatures within the image, and use the source color (determined by Edit->Options->Color Comp Wavelengths*) of the selection point as the class map overlay color. The mean spectra does not depend upon the classification points whereas the difference image does.*
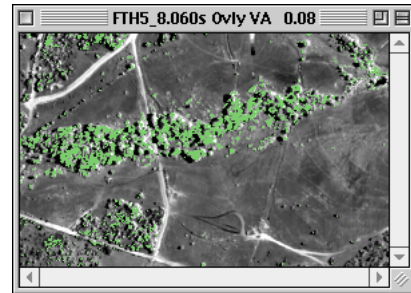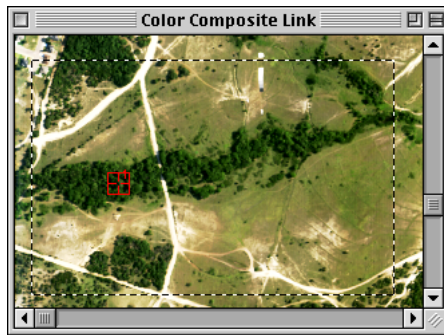
A single selection point within the delineated area shown on the left is used to produce the overlay type classmap appearing to the right.
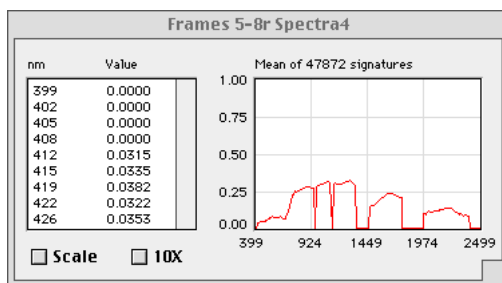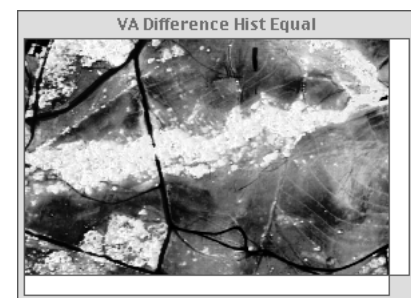
The plot of the mean of all 47,872 spectral signatures in the classification area. Bad bands are zeroed. A plot that includes bad band values can be generated by choosing Enable All *in the* Band List *and then reclassifying. The difference image (shown right) is coded so that the <u>brightest</u> values represent the <u>closest</u> match to the chosen selection point(s) for the given classification method. However, the data attached to the difference image gives the actual metric difference (e.g. radians for* Vector angle*) as the cursor is moved over the image and the* Info Window *(menu* Windows->Show Info*) is active.*
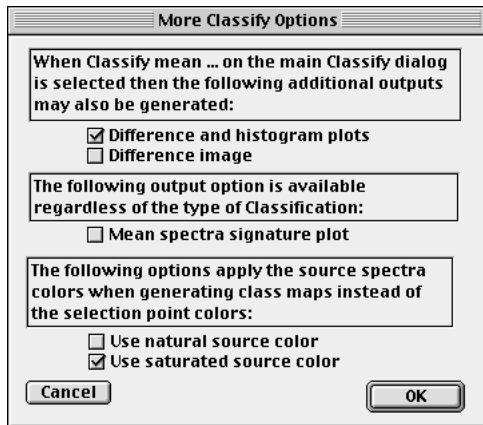
The selected option will produce two plots: The actual value produced by the Method *used in the classification for each pixel and a histogram that plots a relative frequency count of how often a specific difference occured. Any section of each plot can be magnified by dragging a selection rectangle about the area of interest and then choosing the* Image->Magnify->2x *menu. The magnification factor is ignored, instead, the selection rectangle is expanded to full size.*
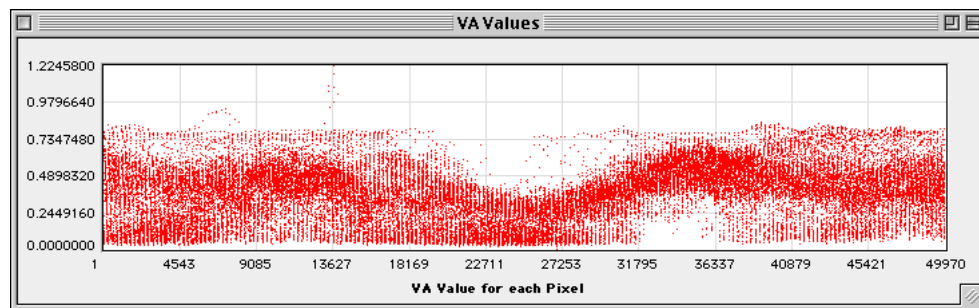
Each of the 49,970 spectral signatures is plotted against the Vector Angle *difference between it and the chosen classification point shown on the previous page. The maximum difference is 1.22 radians. By dragging a selection rectangle anywhere within the plot and choosing* Image->Magnify->2x *a new plot consisting of just that region will be constructed. This may be repeated on the new plot to get even finer detail.*
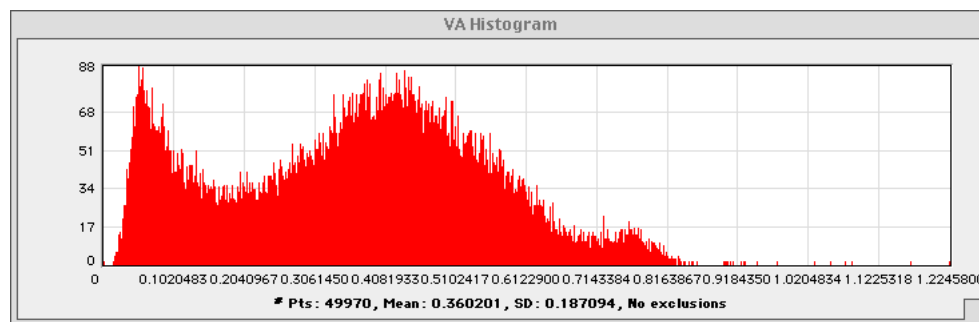
This Histogram *is essentially the same plot plot as the* Difference *plot reordered so that the horizontal axis represents* Vector Angle *differences (in radians) versis how frequently this difference occured. This plot may also be magnified repeatedly.*

Image Products

Each Classify produces an image(s), an ordered set of spectral file names or an ROC plot. The image can be a color representation of the goodness-of-fit or a composite consisting of the color representation overlaid on an image background. The background is automatically chosen to be the Search Domain band face so for example if band number 35 is visible, a duplicate is made and used as the overlay base. The colors of the mask/overlay are the same colors as √ points in the classify list. If a selection rectangle is present on the Search cube face then the classification and resultant image is limited to the rectangle size.

An output image has one or more classification regions depending upon the various options described in the previous section. The default is one region per √ point or √ library spectra. However, if the Classify mean radio button is chosen then the average of all √ points is formed for the source signature for comparison and therefore only one classify region is generated. Shown below is a source image with four selected points, and 2 classification outputs using the default Vector Angle Method with tolerances .08 and .8 radians respectively. Each region color corresponds to the selection point color and can be changed via the Color Selector (see section Changing Selection Point Colors) or Use source spectra color in More Options.

The utility of the classification output can be greatly expanded by using Flicker/Superimpose in the Functions menu or compositing it with another image via Overlay Color or using image Arithmetic. Furthermore, classifications can have labels that dynamically change as the cursor passes over them (see section Dynamic Displays and Plots).
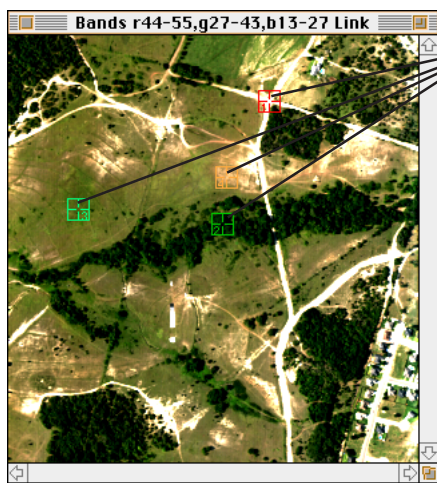
Four selected points on a linked color composite.
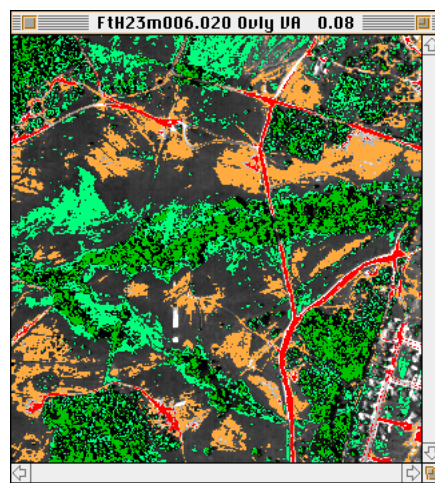
The top right output is the result of using the Vector Angle Method *with a tolerance of .08 radians. The base image cube shows through where the angle exceeded the tolerance. The right image is the same classification re-run with a tolerance of .8 radians which just about forces each pixel into one of the classification regions thus oblitering the underlying image.*
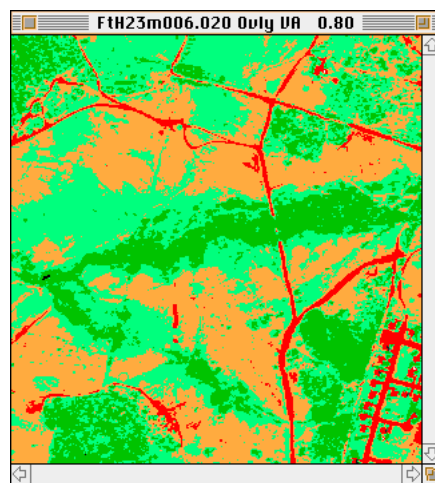
One of the more interesting products of Classify is the generation of a synthetic cube consisting of a linear sequence of classification tolerances. For example, the Classify Options dialog shown below will generate 11 outputs ranging in tolerance from 0 to .1 radians that will be automatically collated into an 11 band <u>synthetic</u> cube also shown below. Besides band scrolling, most of the HyperCube functions that can be applied to a spectral image cube can also be applied to the synthetic cube, such as: Gray & Color mapping, Flips, Rotates, Flicker/Superimpose, Magnify and Filtering. The exceptions are Shifts, Histograms and Statistical Measures.

Vector Angle is the chosen Method.

The number of outputs in the synthetic cube. In this instance the output sequence will be: 0, .01, .02, .03, .04, .05, .06, .07, .08, .09 and .1 radians tolerance.

Starting tolerance of the first output.

If checked then a plot of "hits" versus "false alarms" will be generated.

Ending tolerance for the last output.
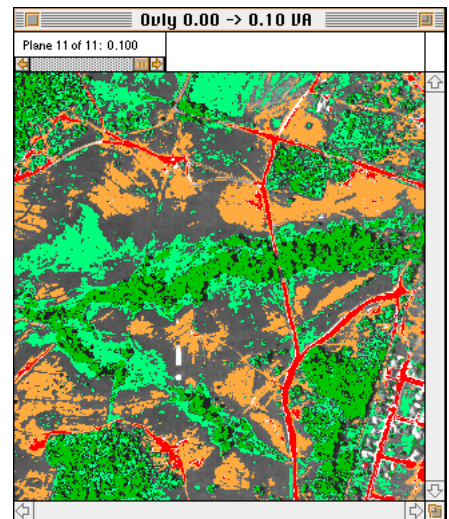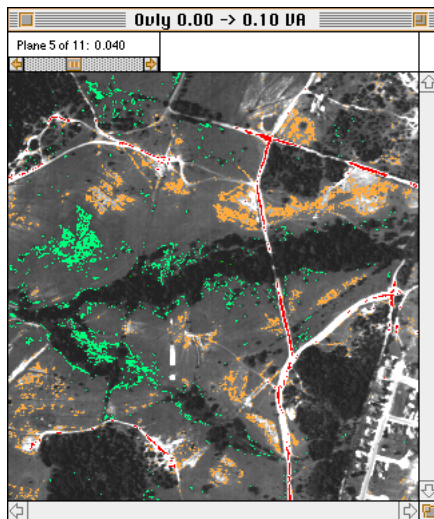
Shown left and right respectively are two views of the resulting synthetic cube. The left one has been scrolled to the fifth output band in which the overlay tolerance is .04 radians while the right view has been scrolled to the last band showing a tolerance of .1 radians.

The classification regions of a classify image can be tabulated by selecting menu Functions->Plots->Histogram when the image is active. The result is list of each class name and the frequency with which it appears in the image. Zero frequency classes are omitted. The list can be selected (normal drag selection and command-C key) and pasted into another program, e.g., Excel®. Below is a portion of a very detailed Lib:File classification and the histogram of the selection rectangle area within the image.
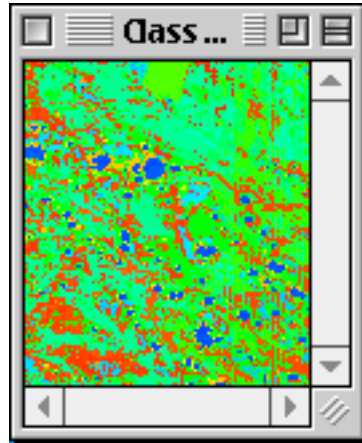

```
#   Count  Name
 5      4  BG-1, asphalt, light
 6      1  BG-2, asphalt, light
13     16  BG-1, roadway, packed
14      1  BG-1, roadway, gravelly
15      1  BG-2, roadway, gravelly
16    186  BG-3, roadway, gravelly
17    165  BG-4, roadway, gravelly
19     36  BG-6, roadway, gravelly
22     11  BG-1, PW3b, sand
23    144  BG-2, PW3b, sand
24     15  BG-3, PW3b, sand
25     39  BG-4, PW3b, sand
26     29  BG-5, PW3b, sand
27     17  BG-6, PW3b, sand
28    112  BG-1, F1c, gravel
29    302  BG-2, F1c, gravel
30      3  BG-3, F1c, gravel
31    343  BG-1, T1c, gravel
32     97  BG-2, T1c, gravel
33     50  BG-3, T1c, gravel
```

At the left is a portion of a classmap generated via a Lib:File *classification. The library contained several hundred signatures. At the right is part of the histogram frequency count listing for this classmap. Each library signature has multiple measurements. The first column contains the index of the library signature in the color key. See sections:* Library Searches *and* Color to Name Editor *for a description of the class name and color association.*

F(Signatures)

Whenever the Classify dialog is active you can construct the eigenvectors and/or orthonormal basis of the selected signatures via menu item Functions->F(Signatures). Either image or library signatures must have been toggled on and Classify each... must be selected. If from a library they are resampled to the enabled bands of the attached image cube. Eigenvectors requires 2 or more signatures while Orthonormal 1 or more. The covariance or the correlation of the signatures may be chosen for the eigenvectors computation. Since the number of signatures is usually much less than the number of bands the resulting eigenvalues and hence eigenvectors have little significance after the N-1 value where N is the number of signatures used. Compare this with the HyperCube menu item Image->Convert Image->Principal Components that uses all of the pixels within a cube (or selection rectangle) to compute the covariance. Orthonormal uses the Gram-Schmidt method and the number of signatures can <u>not</u> exceed the number of bands. Orthonormal also produces the corresponding projection matrix defined as: $A(A^TA)^{-1}A^T$ where A is the column matrix of orthonormal vectors. Both methods generate matrices that are saved to a file and can be used in Utilities->Apply Transformation.

Mathematically, the following matrices are produced for the particular instance of an image cube with 160 of 210 bands enabled and 10 signatures.

Eigenvector matrix (Matrix.ev):   160 rows x 160 columns.
Orthonormal matrix (Matrix.gs):   160 rows x 10 columns.
Projection matrix (Matrix.pm);   160 rows x 160 columns.

Below and on the following page are examples of the Eigenvectors method and the result of applying the matrix to the original image cube using menu Windows->Apply Transformation.
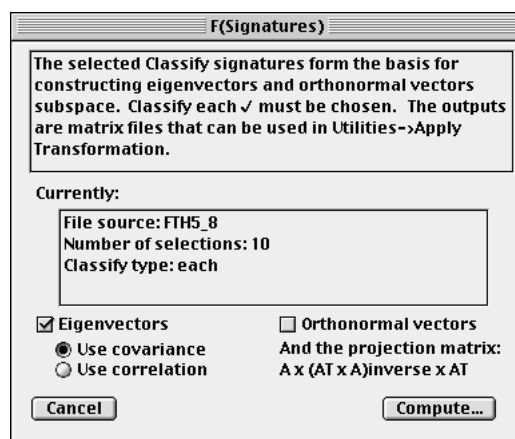
F(Signatures) *dialog set to compute the eigenvectors corresponding to 10* Classify *signatures. The result is a matrix file.*
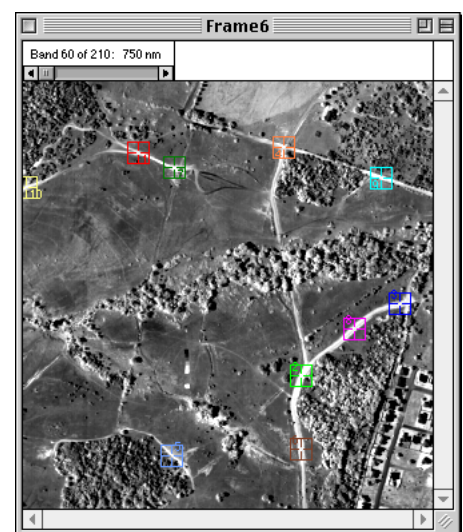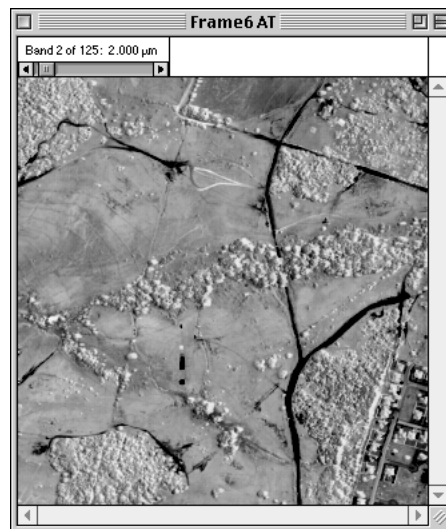

Classify *selection points.*

Frame6 AT
Band 1 of 125: 1.000 μm
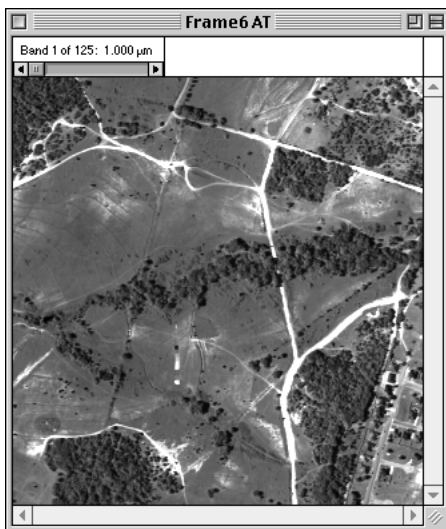
First principal component.

The left and right images are the first two bands of the cube resulting from applying the eigenvector matrix to the original data using <u>Utility Apply Transformation.</u>

Frame6 AT
Band 2 of 125: 2.000 μm

Second principal component.

ROC Curves

One type of classification output is a ROC (Receiver Operating Characteristic) curve. This produces a plot of the probability of correctly classifying a data set (statistically <u>not</u> committing a Type I Error) verses the probability of erroneously classifying a data set (statistically of committing a Type II Error). This relationship is commonly referred to as the "hit rate" verses the "false alarm" rate as a function of a varying decision criteria. In HyperCube the criteria is the tolerance for a given classification algorithm. The validity of whether or not a signature is classified correctly is determined by a previously constructed "ground truth" image mask matching the classification area. A mask pixel of zero means there shouldn't be a match and a nonzero pixel means there should be. During the ROC computation as each image point is classified (matched against a given spectral signature) a tabulation is maintained of whether an image point was classified (within tolerance for the given algorithm) and if the classification was correct according to the "ground truth" image mask. Each tolerance value produces one tabulation. A sequence of linearly spaced tolerances (~250-500 values) produces tables that are then normalized to form the ROC plot.

The ROC option is available when Classify mean of √ points is selected in the main Classify window along with Search Domain set to File and the Generate a sequence in Classify Options checked. It is similar to generating the classification image sequence previously described. After checking the ROC curve box the ROC Mask... button is clicked to select the ground truth image mask from a popup list of images. This image must spatially match the classification area used in generating the ROC sequence of classifications. Note that "Classify mean of √ points" always results in a single classification region so the summation is over a single class.

Listed below is a representative scenario for generating a ground truth mask from an image cube and then forming the various ROC plots shown on the following pages.

1. Choose several image points representing the same classification feature.
2. Select one or more neighborhoods around each point (to form a good average). If these points exactly delineate the ground truth then click the Form Neighborhood Mask Image button and Classify mean of √ points and continue at step 7.
3. Choose an algorithm such as Vector Angle and Classify mean of √ points.
4. Choose Classify Options and set up a sequence of ~15 outputs and Color Image.
5. Click Classify. Choose the output within the sequence of images that best represents ground truth for this feature. Select Convert Image->Image to Mask menu to extract the mask. This mask can then be further edited (e.g. PhotoShop®) to give it more validity.
6. Clear all of the selection points that were used to generate the mask and select a new point from the cube that represents the feature in the mask. Or, open a library and select a signature from the library list (see section Library Searches) and select Library as the Signature in the Classify window.
7. Choose an algorithm from Methods in the Classify window. Select Classify Options, check Generate sequence and ROC curve and then ROC Mask... . Select the mask image generated above from the popup (this only needs to be done once). Return to Classify Options and enter the number of outputs (~250) and a broad range of tolerances for the chosen algorithm. Return to the Classify window.
8. The Classify button will change to ROC Curve. Click it to generate the plot.
9. Repeat steps 7 and 8 for each algorithm. Step 6 may also be repeated to see its effect.

The mean of six points representing a typical tree canopy was used for this image mask that was extracted from a <u>sequence</u> of 15 classifications using Vector Angle.
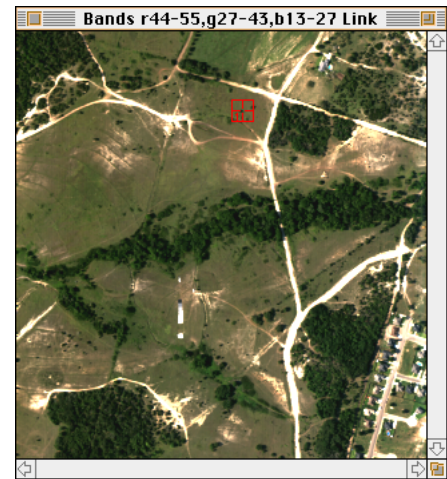

The selected image point on this linked color composite was the classification signature to be matched when generating the ROC *plots shown below.*
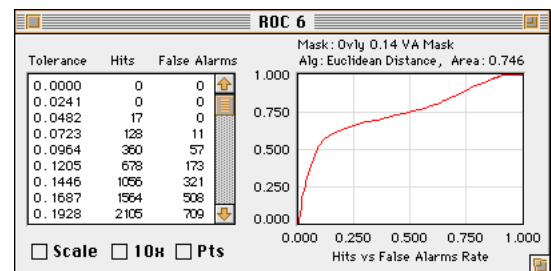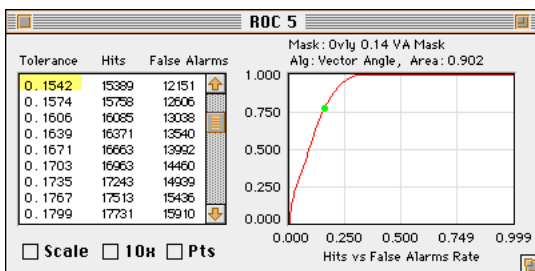

From above clockwise are the ROC *plots corresponding to* Vector Angle, Euclidean Distance, Fourier Phase and Derivative Difference *respectively. The vertical axis is the hit rate and the horizontal axis is the false alarm rate. The greater the area under each curve the more discriminating the algorithm. Ideally, the* Hit Rate *would approach 1 while the* False Alarm Rate *stays near zero producing an area close to 1.*
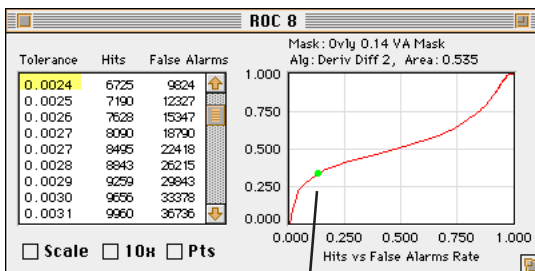

Clicking within the graph displays a green dot and highlights the corresponding tolerance value. Conversely, clicking within the scrolled list shows the corresponding plot position.
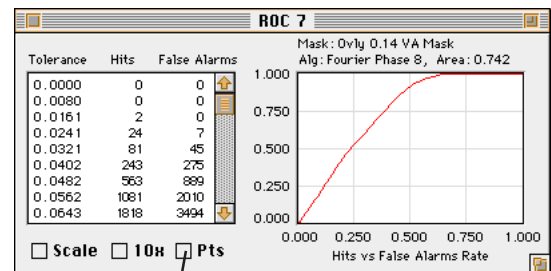
The Pts *checkbox switches the plot from continuous to a discrete scatter plot which the* ROC *actually represents.*
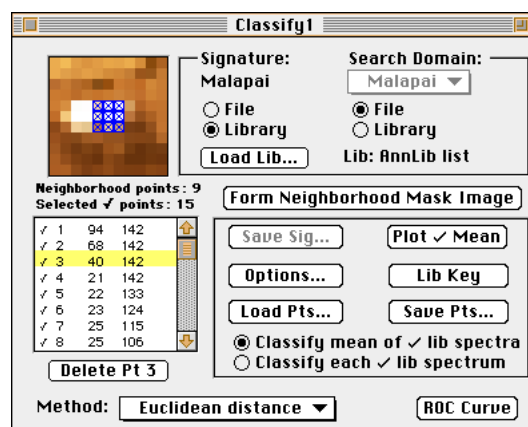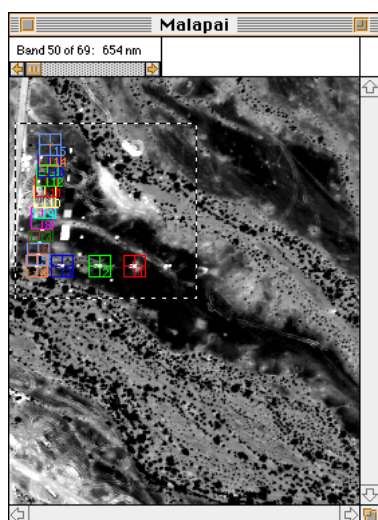
Fifteen selection points along with various neigborhood points were used to generate the resulting selection mask shown to the right by choosing Form Neighborhood Mask Image *in the* Classify *dialog above (see section* Classify Function*). Each target point represents the same physical material. The library spectral signature of this material was used in the classification to produce the four* ROC curves *shown below. Note that the classification area and the mask selected for the* ROC curves *must spatially coincide which in this case was automatic.*
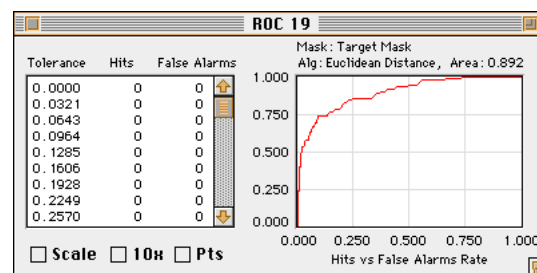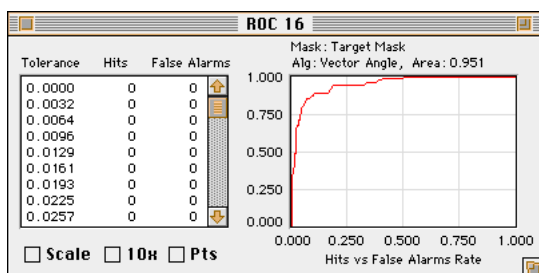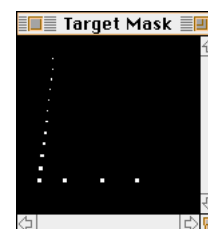
These plots correspond to the same four methods shown on the previous page that were used in discriminating the tree canopy. Here they give a quantitative measure of target verses background discrimination.
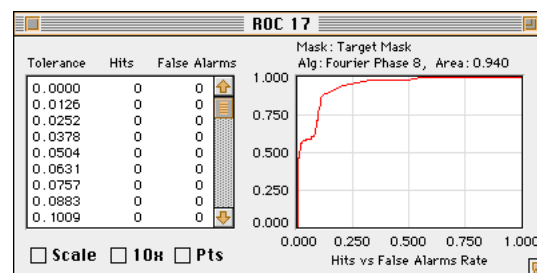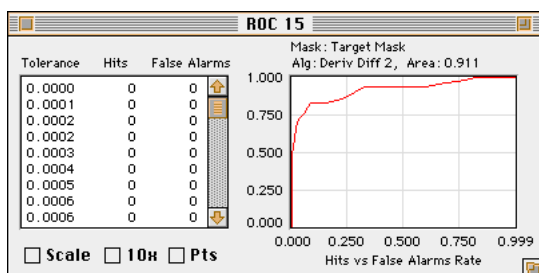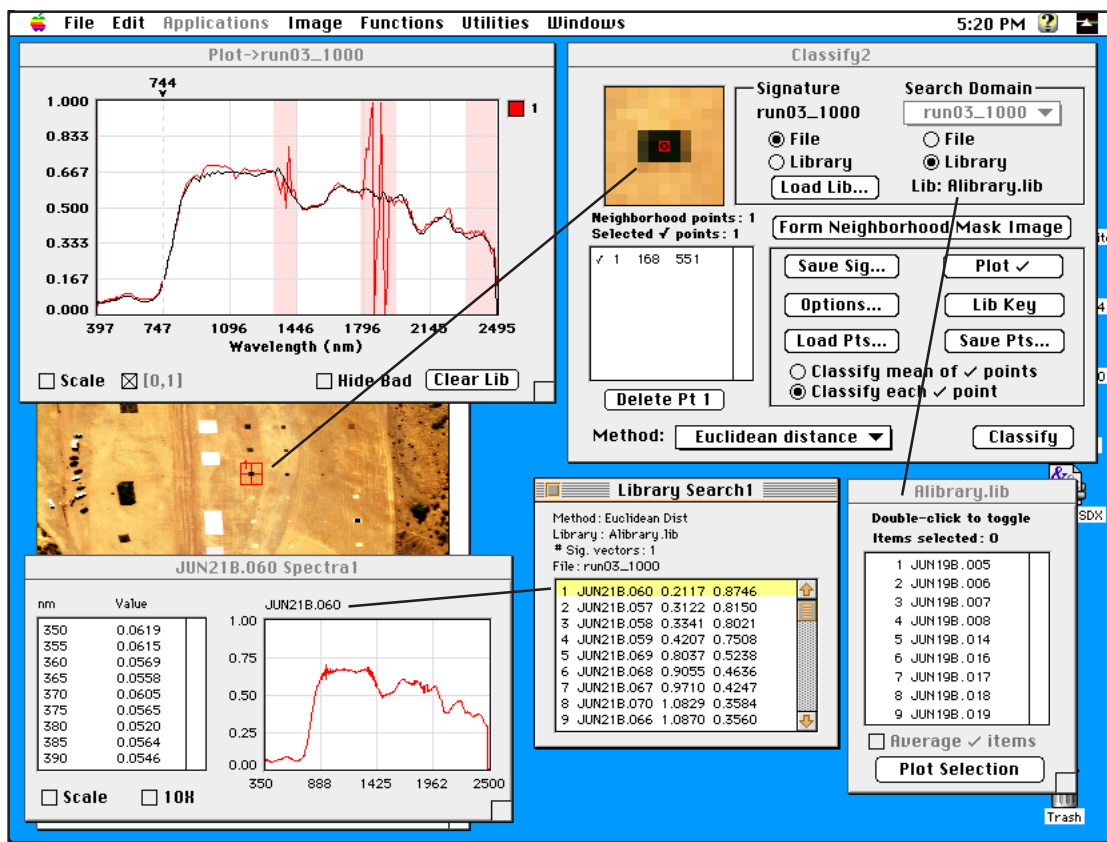
Library Searches

Choosing Library as the Search Domain produces a list of library signatures that best match the criteria selected by the user. The list is ordered in decreasing closeness with a numerical value derived from the classification Method. The numbers are only meaningful within the same method and are most useful in detecting a cluster of matching signatures, i.e. items that are physically similar. The spectrum of any listed signature can be plotted in a spectral signature window by double clicking it and can be added to the spectral plot window by dragging its graph (see the Superimposing Signatures section).

A library is loaded by choosing the Library... button in the Classify window. Once loaded the name will replace Lib:(none) and the Library option radio buttons under Signature and Search Domain will then become available. Loading a library connects it to the classification process whereas using menu File->Open As..., and selecting Format: Library does not and that library remains independent. However, signatures from the later may also be dragged onto the spectral plot window.

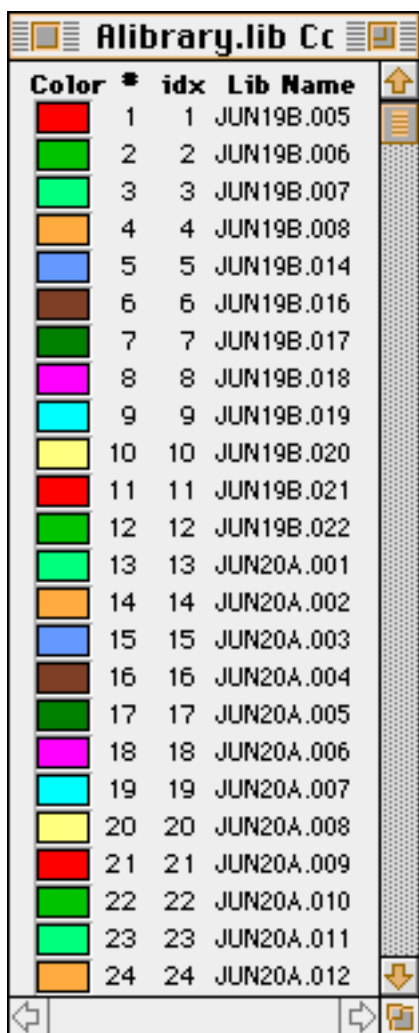The steps to perform a typical library classification are as follows:

1. Load the image cube file and select one or more (if averaging) image points.
2. Choose Classify... from the Functions menu.
3. Click the Library... button to load a spectral library.
4. Select (√) one or more cube points from the Classify coordinate list.
5. Click the Signature: File and the Search Domain: Library options.
5. Choose a classification from the Method popup menu.
6. Click the Classify button. A list(s) of matching library signatures will appear.

The Library Search window lists the library signatures matching the selected image cube file point. The top signature in the list has been superimposed upon the Plot window.
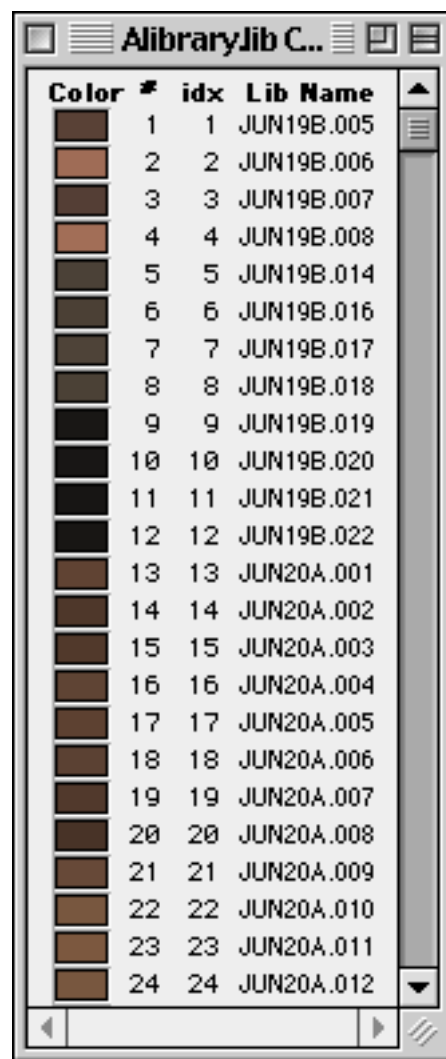
The opposite of the previous example is also possible: i.e., choosing one or more library signatures and generating a classification output (Lib:File search). The region colors are then taken from the patches in the Color Selector as follows: the lowest √ library signature index corresponds to the first color patch, the next higher √ library signature index to the second color patch and so on (see right hand example below for an alternate method of defining the color correspondence). In the same manner that the selection point color correspondence can be chosen, so can the library signature color correspondence by choosing menu Edit->Options->Color Selector. The color to name correspondence can be displayed by clicking the Lib Key button in the Classify window. This generates a palette like image of the correspondences. This can be saved like any other image. Shown here is a typical output. To select (toggle on) a complete library, click on the library window and then choose Select All from the Edit menu. Up to 1024 signatures can be selected. To deselect (toggle off) all choose Edit->Clear menu or double click individual selected spectra. Section Dynamic Displays and Spectral Plots describes how to change both the color and the associated name after a Lib:File classification has been made.

If Lib:File match lists is checked (default) in Classify->Options then two lists are generated in addition to the classmap. One contains a list of the library selections that are in the classmap and the other list contains the library selections that are not in the classmap. Toggling a list item will display its corresponding spectra. Empty lists are not generated.


At left is a portion of 412 color to library signature correspondences after Select All *was chosen in the* Edit *menu to toggle-on all of the library entries showing the default colors. At right is the same library color coding after* Use source spectra color *has been selected in* Classify: Options: More Options.

Dynamic Displays and Spectral Plots

Each time a classification output image is created a list is generated that associates the colored regions with the selection point number or library signature name corresponding to that region. Then, as you move the cursor over the classified image the name appears as a small label next to the cursor. The labelling font and its position can be changed by choosing menu Edit->Options->Labels. This brings up the submenu displayed at the right. If a classification image is saved the correspondence list is also automatically saved using the name of the image concat-

enated with a ".c2n" extension and if the image is later loaded the correspondence list is also loaded and associated with that image. The correspondence between the color of a region and the label name that appears is completely editable by using the Utilities->Class Map Editor described later.
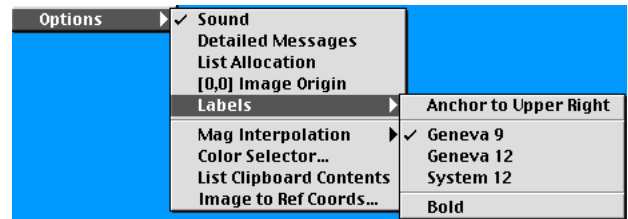
Below are two examples of classificaion outputs showing the region labels for a specific cursor position.
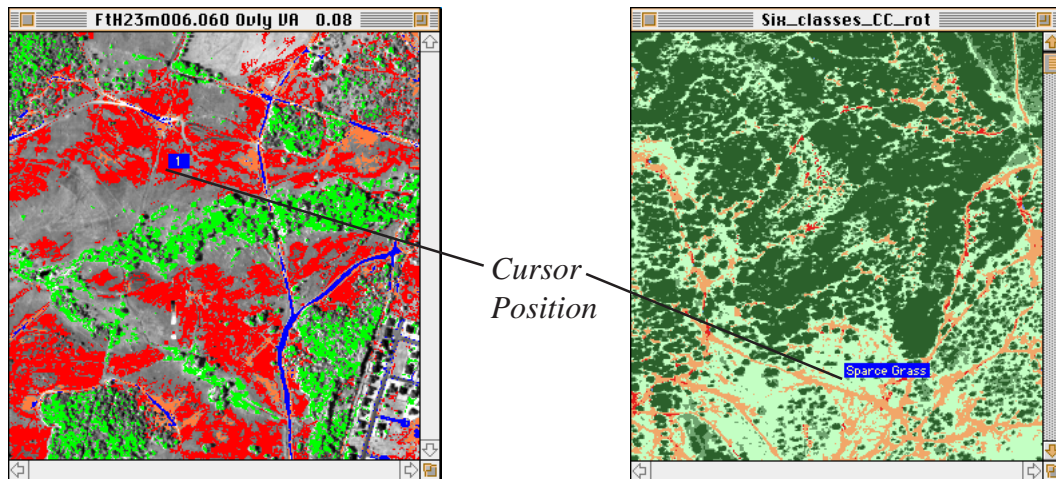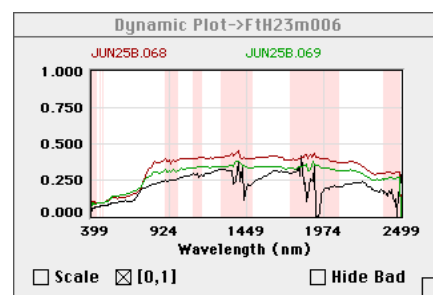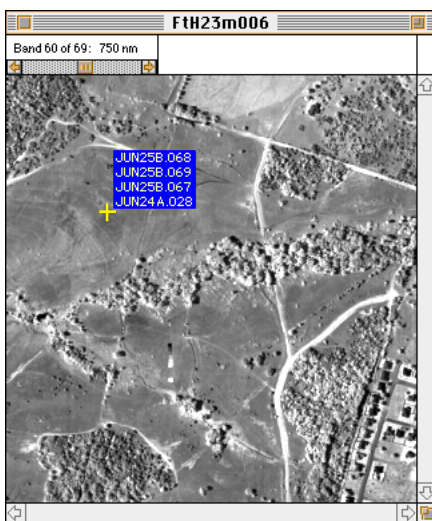

On the left is the output image from a 4 point selection File:File *classification showing point #1's region label.* Vector Angle *was used with a tolerance of .08 radians. To the right is the classification output of a different image cube resulting from a* Lib:File *search. The label and color of either image can be changed via menu* Utilities->Class Map Editor.

A dynamic display similar to a classification output image can be generated when performing a File:Lib search. Normally, this consists of selecting an image cube point and classifying it against a library of signatures producing an ordered list of best matches. However, if the Enable Dynamic Display check box in the Classify Options dialog is selected as shown here then the match vector is chosen from the cursor's position on the image cube. A label containing the best matches will be computed and dis- played as the user moves the cursor. The dy- namic search remains in effect until any mouse click in the Classify window. It must be restarted by clicking the

Classify button. If a color composite Link image of the cube image has been created then the dynamic display will also appear when the cursor is moved over the linked image.

The dynamic display of 4 library spectra that most closely match the image cube spectrum at the cursor position. The graph is the corresponding dynamic spectral plot of the top 2 library spectra (red and green) along with the cube plot in black. In this case the Method *chosen was* Vector Angle.

A dynamic display of the current image cube vector position can be shown by selecting Spectra (Dynamic) from the Plot submenu. As the user moves the cursor over the image cube or linked color

composite the plot is automatically updated. The data is read from the image disk file and works best when the cube file is in BIL or BSQ format. If the user has enabled Dynamic Display in the Classify Options dialog, described previously, then the two closest library spectra will also be plotted on the same graph. Similar to the static Selected Points plots, you can 'Scale' and 'Hide Bad'.
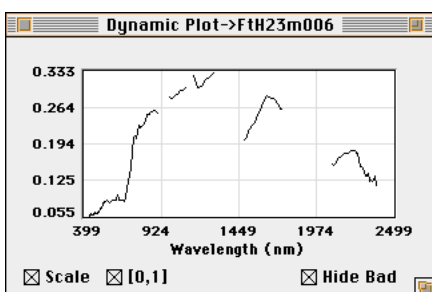
Dynamic spectral plots of the same cube vector point with and without the optional check boxes selected.
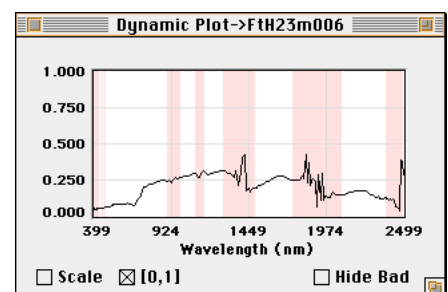
Class Map Editor

The correspondence between the color and name of an image classification can be modified by using Utilities->Color to Name Editor. You can do this after the image is generated or at a later time as long as the classification image has an associated *.c2n file. A classification image is a true color im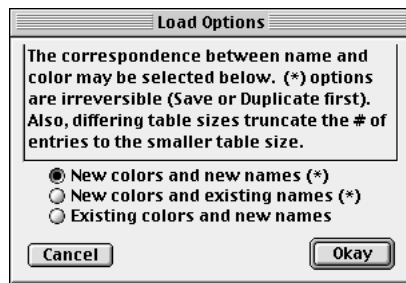age where each pixel is either a gray value (including black if it's a mask) or a specific color. Associated with each color is an ASCII string comprising the name that will be displayed whenever the cursor is over that color or no display when over a gray (image) pixel. The Editor allows changing that association so that a color or group of colors may be "renamed" or the color itself changed with a corresponding change in the image. For example: a classification region titled "Water" and represented by a light cyan color can be changed to "Shallow Pond" with a medium blue color. A single colored region can be changed to any other color (not gray) as long that color is not already used for a different region. You can choose multiple items using the shift key for contiguous and the command key for non-contiguous selections (Windows version uses the Ctrl key for non-contiguous). As each item is chosen the corresponding class on the image will flicker in white. So, as multiple selections are made they produce a larger and larger composite region. The Apply button will consolidate all of the selected items into the displayed name and color and change the image appropriately. Apply is not reversible. Therefore, it is best to make a duplicate (Edit->Duplicate) of the class map image before editing it.
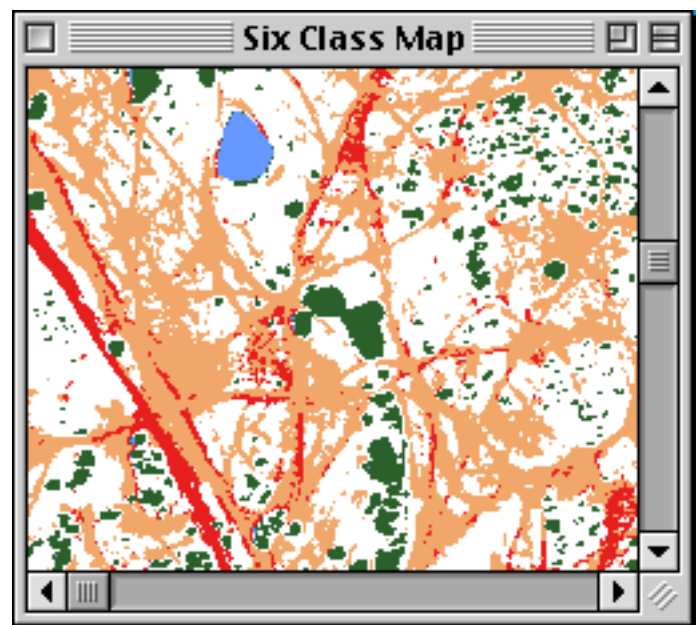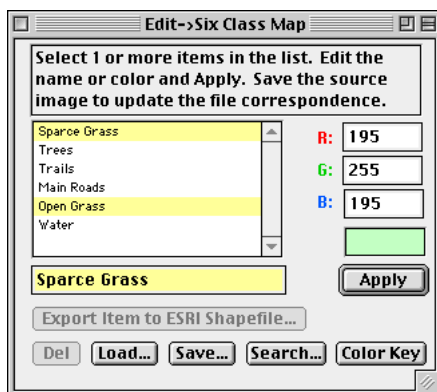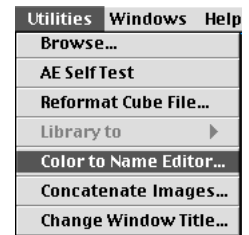
The Editor window on the left shows 2 highlighted items that have been selected and are about to be consolidated into a single class with the name and color shown. The classmap image on the right shows where these 4 items are located by flickering them in white. When Apply *is clicked both items will have the color:* {195,255,195} *and the item list will be updated. The* Search *button displays a dialog allowing you to enter a text string and every list item containing that string will be highlighted as if manually selected. The* Export... *button saves a single item in the ESRI Shape File format (see section* ESRI Shape Files*).*

Scrollable List	For each classification region the associated name and color. Click to select, edit name and color followed by Apply or return. Multiple lines may be selected via shift click and command click.
R:G:B:	The color components as each is selected. Edit as desired. The color patch below shows the composite.
Apply	Change the selected list name to that of the edit field and color and update the image. Hitting return key is same as Apply.
Delete	Remove the selected list item and replace its color with black.
Color Key	Generate a legend type image (see section Library Searches).
Load/Save	The color to name correspondence may be saved without having to resave the image. It must be saved as '"name".c2n. A previous (e.g. master) may be loaded to overwrite an original correspondence. This is an ASCII file and is also compatible with Edit->Options-> Color Selector. See section Changing Selection Point Colors.

Load Options

The correspondence between name and color may be selected below. (*) options are irreversible (Save or Duplicate first). Also, differing table sizes truncate the # of entries to the smaller table size.

● New colors and new names (*)
○ New colors and existing names (*)
○ Existing colors and new names

[Cancel]   [Okay]

Various combinations of the current and new colors and names may be selected.

Search	This displays the dialog shown below and is used to select items based upon their common names. Each item containing the search string will be highlighted. No modifications are made until Apply is clicked.

Search Color to Name Editor

Enter a search string below. Each line item within the Color to Name Editor containing this string will be highlighted. Clicking Apply will collapse all of the selected entries to the displayed name and color. Apply is not reversible.

☐ Case sensitive ☑ Preclear selections

Alphalt

[Cancel]   [OK]
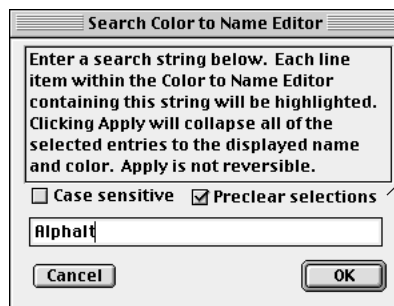
If Preclear is checked then any existing highligting is cleared else items matching the search string are added to currently highlighted items.

Export Item...   Convert a <u>single</u> item into a shape file as specified in the ESRI Shapefile Technical Description - July 1998. Brings up the user dialog shown below presenting a number of options.


Save files: Solicits a file name using the item as the base name and produces the three files listed. The .shp extension is a must, the other two are automatic.

Plot shape: Generates a vector plot of the result at 2 times size. This plot will exactly match a 2 times magnified image of the class map. See sections Magnify and Flicker/Superimpose.

Flip output: Invert the y coordinate so that the origin is at the upper left instead of the image's lower left.

Transform: Convert shape file coordinates to the transformation specified by Image to Reference Coordinates.
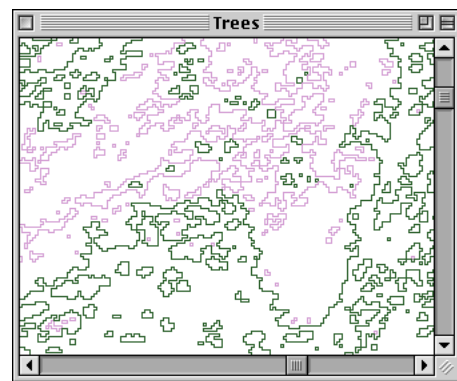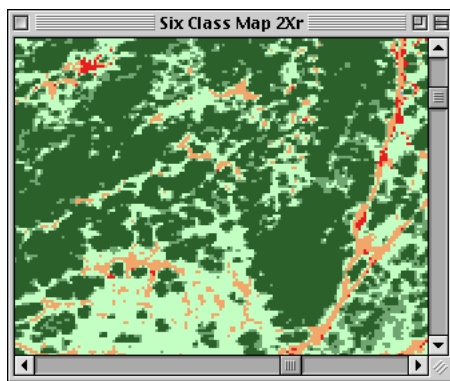
The left image is a 2x magnified view of a six class map. The main dark green regions are the selected item in the Class Map Editor *that was* Exported *to a shape file. The right image is the vector plot output showing the class outline in green and any interior voids in the complement color, magenta.*

Spectral Calibration

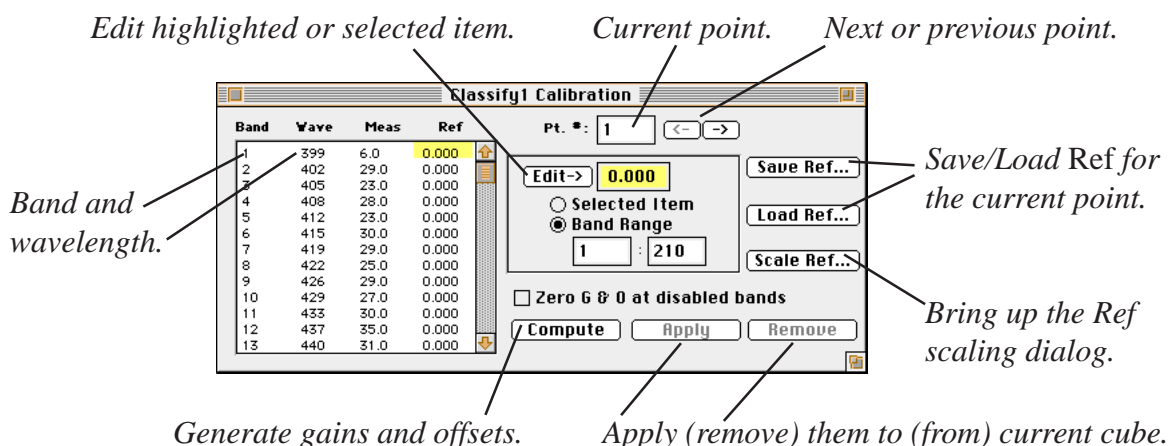   HyperCube has the ability to empirically calibrate a cube against a set of two or more reference spectral vectors. The reference may be from an external file, another cube or a library. Calibration uses the same point selection and neighborhood toggling as is done for a Classification. It can be invoked whenever a Classification window for a given cube is active by choosing Functions->Calibration. Calibration produces a gains and offsets file that may be saved or immediately applied so that you can see the effect on a classification or plot by toggling it on and off.
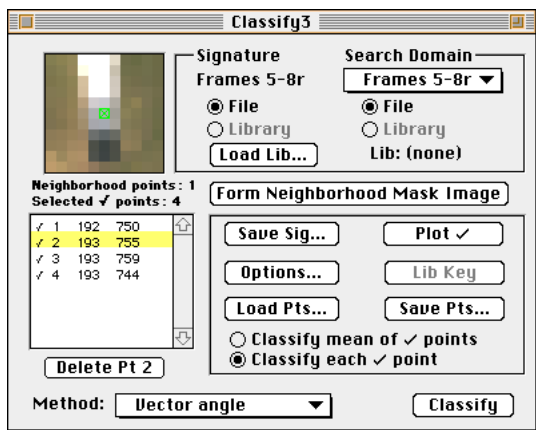
   The idea behind calibration is simple, try to compute a separate linear transformation, gain (or scale) and offset, for each band that will convert that band's pixel values to the same values in a reference set. The assumption is that the difference between the measured and reference spectral values is closely represented by a linear change. In practice this isn't exact so more than two spectral vectors (selection points) are used in a least square solution.

   A typical calibration scenario is as follows: Suppose there are four panels in the source cube that have known spectral reflectance signatures (somewhere else). Then choose four selection points over these panels and toggle them on as if doing a classification. You may then choose additional neighborhood points around these if an average spectral signature for each panel is desired. Bring up the Calibration dialog as stated above. The Meas column will be the measured spectral vector for the given point and the Ref column will be the desired reference spectra and initially will be zero. The next step is to put data into the Ref column that you want the Meas value to be transformed into. This could be done by manually entering the data as described below (very tedious for hyperspectral data!). However, the easiest method is to drag an existing reference spectral signature plot over top of the calibration window list. It will be automatically interpolated from its wavelengths to the cube's wavelengths (see section Plotting Signatures). The signature plot can come from a library list (see Spectral Libraries) or from another cube. The later is extracted by loading a second (reference) cube, selecting a point, opening a classification window, toggling that point on and then clicking the Plot √ button. The resulting signature plot can then be dragged onto the calibration window. The final method of entering Ref data is to read it from a previously generated file. Its format is ASCII tab delimited text pairs of wavelength and value. Note that it is not defined by a band number. The wavelengths are used to properly interpolate the values into the cube's wavelengths. If the cube's spectral range extends beyond the Ref values then the corresponding Ref data is zero filled.

Edit highlighted or selected item.   *Current point.*   *Next or previous point.*

Band and wavelength.

Save/Load Ref *for the current point.*

Bring up the Ref scaling dialog.

Generate gains and offsets.   *Apply (remove) them to (from) current cube.*

The various Calibration Dialog controls are defined below:
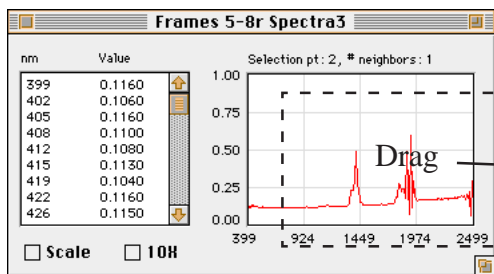
Scrollable List: Contains four columns listing the Band, Wavelength, Meas value
                    and Reference value for the given Pt. #. Clicking on a
                    Meas or Ref column item allows that item to be edited, see Edit button.

Pt. #:   The current selection point number shown in the list. May be manually
                    edited or incrementally changed with the adjacent arrow buttons.

<- ->:   Step to the previous or next selection point.

Edit->:   Enabled when a Meas or Ref item is highlighted (clicked on). The high
                    lighted item will be changed to the number in the edit field. Works in
                    conjunction with the two radio buttons described next.

Selected Item: If on then only the selected (high lighted) Meas or Ref item will be
                    changed. Mutually exclusive with Band Range.

Band Range: If on then all of the items within the Band Range specified will be
                    changed to the same given value. Mutually exclusive with Selected
                    Item.

Save Ref...   Displays a standard Mac save dialog and saves the current point Ref data
                    as ASCII tab delimited wavelength, numeric value pairs.

Load Ref...   The opposite of Save, loads a previously defined Ref file and interpolates
                    as necessary to overwrite the current point Ref data.

Scale Ref...   Brings up a dialog that permits you to enter a numeric value that will
                    immediately be multiplied times a specific Ref column or all Ref columns.
                     A convenience to change the range of the Ref data.

Zero G & O...   When checked the gain and offset computed for each underlined disabled band
                    is set identically to zero. When these gains and offsets are then applied
                    to a cube the disabled band's spectral values will then be zero.
                    See section Band List for a description of disabled bands.

Compute   This button becomes active when at least two Classify points are selected.
                    When clicked the gains and offsets are computed and shown in a separate
                    data list window that can be saved via Save in the File menu.
                    Also generated is a data list showing the corresponding residuals from the
                    least squares solution.

Apply   Becomes active after a Compute and immediately applies the gains and
                    offsets to the current cube meaning that any plots are automatically
                    updated and subsequent classifies will accommodate them.

Remove   Remove any gains and offsets that may be in effect for the given cube.

Classify3

Signature — Frames 5-8r
Search Domain — Frames 5-8r ▼
◉ File   ◉ File
○ Library   ○ Library
[Load Lib...]   Lib: (none)

Neighborhood points : 1
Selected ✓ points : 4

✓ 1	192	750
✓ 2	193	755
✓ 3	193	759
✓ 4	193	744

[Form Neighborhood Mask Image]
[Save Sig...]   [Plot ✓]
[Options...]   [Lib Key]
[Load Pts...]   [Save Pts...]
○ Classify mean of ✓ points
◉ Classify each ✓ point

[Delete Pt 2]
Method: [Vector angle ▼]   [Classify]

Classify2

Signature — data8a
Search Domain — data8a ▼
◉ File   ◉ File
○ Library   ○ Library
[Load Lib...]   Lib: (none)

Neighborhood points : 1
Selected ✓ points : 4

✓ 1	186	367
✓ 2	184	375
✓ 3	180	384
✓ 4	177	392

[Form Neighborhood Mask Image]
[Save Sig...]   [Plot ✓]
[Options...]   [Lib Key]
[Load Pts...]   [Save Pts...]
○ Classify mean of ✓ points
◉ Classify each ✓ point

[Delete Pt 2]
Method: [Vector angle ▼]   [Classify]

Four selection points from cube Frames 5-8r *have been chosen as reference values. Point #2's spectra is shown below and is being dragged over the corresponding point #2 listing of* data8a*'s calibration window*
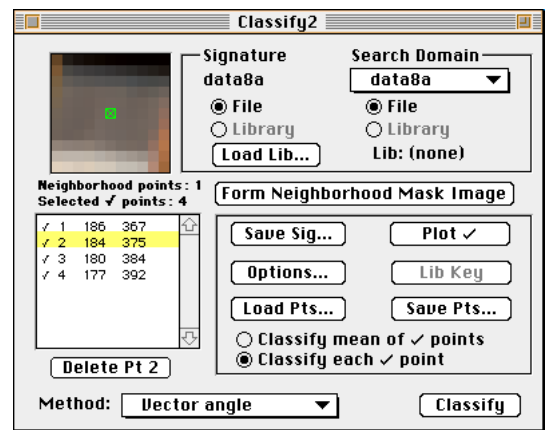
These are the four points of data8a *that have been selected as the matching spectral vectors to the four reference vectors of* Frames 5-8r *along with the* data8a *calibration window below.*

Frames 5-8r Spectra3

nm	Value
399	0.1160
402	0.1060
405	0.1160
408	0.1100
412	0.1080
415	0.1130
419	0.1040
422	0.1160
426	0.1150

Selection pt: 2, # neighbors: 1

Drag

☐ Scale   ☐ 10X

Classify2 Calibration

Band	Wave	Meas	Ref
1	434	0.126	0.000
2	440	0.127	0.000
3	445	0.130	0.000
4	451	0.125	0.000
5	457	0.129	0.000
6	462	0.129	0.000
7	468	0.130	0.000
8	473	0.130	0.000
9	479	0.131	0.000
10	484	0.130	0.000
11	490	0.129	0.000
12	495	0.130	0.000
13	501	0.132	0.000

Pt. #: [2] [<-] [->]
[Edit->] [   ]   [Save Ref...]
◉ Selected Item   [Load Ref...]
○ Band Range
[N/A] : [N/A]   [Scale Ref...]
☐ Zero G & O at disabled bands
[Compute] [Apply] [Remove]

Spectral plot of point #2 of Frames 5-8r.

Point #2 of data8a*'s calibration window before the drag operation.*

Classify2 Calibration

Band	Wave	Meas	Ref
1	434	0.126	0.123
2	440	0.127	0.121
3	445	0.130	0.121
4	451	0.125	0.118
5	457	0.129	0.117
6	462	0.129	0.116
7	468	0.130	0.114
8	473	0.130	0.114
9	479	0.131	0.115
10	484	0.130	0.114
11	490	0.129	0.112
12	495	0.130	0.112
13	501	0.132	0.111

Pt. #: [2] [<-] [->]
[Edit->] [   ]   [Save Ref...]
◉ Selected Item   [Load Ref...]
○ Band Range
[N/A] : [N/A]   [Scale Ref...]
☐ Zero G & O at disabled bands
[Compute] [Apply] [Remove]

Data2

Residuals of image: data8a

1	+0.0198
2	+0.0194
3	+0.0206
4	+0.0198
5	+0.0180
6	+0.0188
7	+0.0205
8	+0.0187
9	+0.0184

Data1

Gains and Offsets of image: data8a

1	+1.3375	+0.1014
2	+1.3738	+0.0976
3	+1.3700	+0.0974
4	+1.3684	+0.0945
5	+1.3883	+0.0887
6	+1.3708	+0.0893
7	+1.3728	+0.0881
8	+1.3567	+0.0886
9	+1.3549	+0.0867

Point #2 after the drag operation. The spectral plot values have been interpolated and placed into the Ref *column. This is repeated for each of the remaining points followed by the* Compute *button which then produces the* Gains and Offsets *values along with the* Residuals *for each band. These may be applied immediately (*Apply*) to* data8a *or saved (*File->Save*).*

Saving Images

HyperCube saves images as JPEG, NITF, Pict, .SUN, TIFF, Raw or Multiband depending upon the File menu item chosen and on the internal format of the image. Any single image or image band can be saved as Pict (the intrinsic Mac image type, Windows version is TIFF) by making the image window active and choosing File->Save. If the image has not as yet been saved a standard save file dialog with the suggested name will appear asking for confirmation if the file already exists to overwrite it. Subsequent saves of the <u>same</u> image will perform the save directly automatically replacing the old image with the same name. However, if the image has been modified since the last save the confirmation dialog will again be presented. To save a specific band, scroll the cube to the band, choose Dupli-cate from the Edit menu and then Save. Images in Pict format are loaded by the Open function in the File menu (see the Loading Images section).

File->Save As... is used to save images in JPEG, NITF, Raw, SRF, TIFF or Multi-band format. If NITF, SRF or TIFF is selected from the Format popup then no other dialog is required. The Pict format is also included for convenience. Raw is used to save a single image and allows specifying a rectangular subset and the pixel type. If the loaded image has attached data (e.g., Retain source data was checked in the original load dialog) then the attached data values are saved instead of the 8 bit display values. Selecting Raw displays:
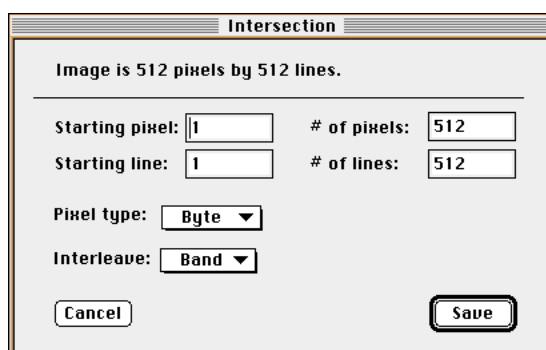
The Save As... Raw *dialog defining the file format for a true color screen dump image saved in* Band *interleave order. Because it is true color it will be saved as a 3 band cube.*

The dialog edit fields are defined as:

Starting pixel:	First pixel defining the image rectangle to save. These first 4 edit fields are preset to the current dimensions of the image.
Starting line:	First line to save.
# of pixels:	How many pixels per line to save.
# of lines:	Corresponding number of lines to save.
Pixel type:	Either byte, short, float or long.
Interleave:	If the image is true color then this popup will be enabled to allow the image to be saved as a 3 band cube with the specified interleave.

The Save As... function is also used to save an image cube as a multiband image. The dialog is similar to the previous Raw dialog with the addition of two more edit fields and a check box to selectively dump only bands whose status is Enabled in the Bands List. Unlike the Raw dialog, the Interleave popup is always enabled, not just for true color. Save As... also generates a new *.hdr, and for multiband a new *.wvl, file.
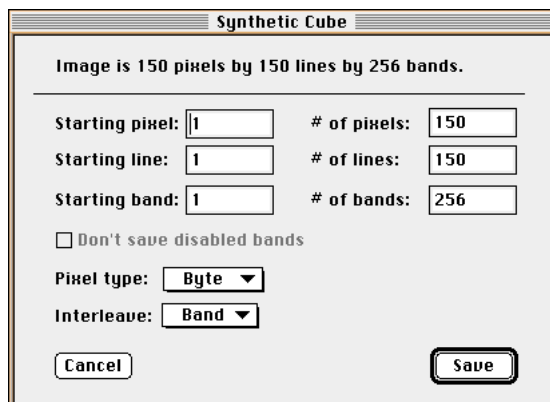
An entire 256 band image cube is specified with no subsetting. The grayed-out appearance of the check box indicates that there are no disabled bands in the Band List.

The additional dialog items are defined as:

Starting band: The first band of the cube to start saving. The default is the first band.

# of bands: How many bands to dump beginning at the Starting band. The default is the number of bands in the image cube.

Don't save dis... If this check box is set then disabled bands specified in the Bands List will be skipped. This option is unavailable if all bands are enabled.

Note: To change the format of the multiband <u>file</u> use the Reformat Cube File... function in the Utilities menu. It is much more general and operates directly on the file pixels and not on remapped images that have been loaded into memory. However, it gives the same result if the file pixels are only 8 bits since no remapping would have been necessary.

The last type of image save is Save Bands... and becomes available in the File menu when an image cube window is active. Its purpose is to save each band of the cube as a Pict file forming a set of sequenced images that can be made into a QuickTime™ movie using a program such as Adobe Premiere™ or ConvertToMovie™. The user is presented with the standard save file dialog showing a suggested base name for the images. This name will be concatenated with the band number (e.g. name.001) as each image is automatically saved. A dialog will appear showing the progress of the save. To prevent clutter it is best to select a separate folder for the images. It is also a good idea to embed a spectral identifier in the cube before saving by choosing Embed Wavelengths from the Image menu. Only cube bands loaded into memory are saved so a very large amount of RAM needs to be allocated to HyperCube in order to contain the entire cube.

Printing

Any HyperCube window or underlying image can be printed to the currently chosen printer using the functions File->Page Setup... and File->Print. These bring up the standard <u>Mac</u> or <u>Windows</u> dialogs with options usually specific to the type of printer. The default is to print the contents of the front most window with its name below centered on the page. Print Image prints the entire image regardless of what is shown within the window whereas Print Window limits itself to just the way the window appears including any operations being performed on the window such as gray mapping or filtering. Print Image will be replaced by Print Text if the window contains any ASCII lists. Whether window or image, the output will always be resized to fit within the print area.
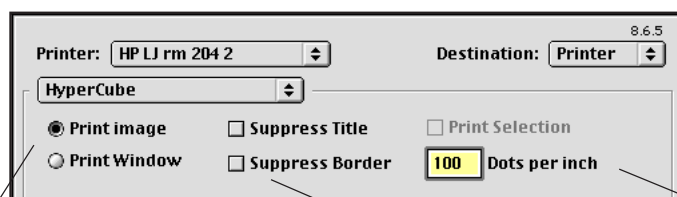
<u>Macintosh</u> *Print Dialog details*

If a selection rectangle has been drawn then you have the option of just printing its contents.

Selects either print the entire image or just that portion shown in the current window.

You have the option of omitting a title or border.

The desired output resolution. However, the image will always be scaled, if necessary, to fit the page.
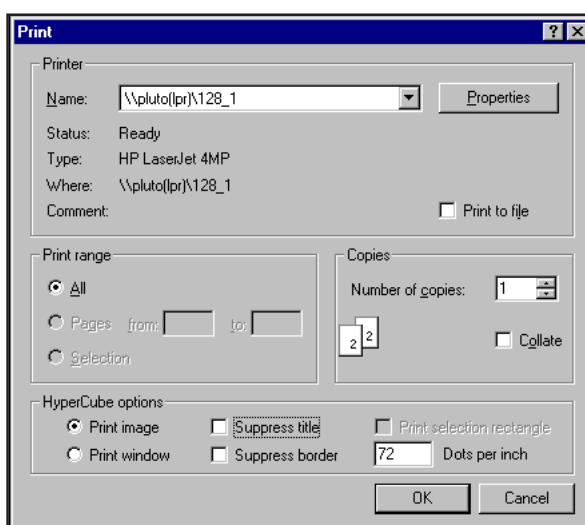
The corresponding Print *dialog in the* <u>Windows</u> *version. The HyperCube options appear at the bottom.*

Utilities

Reformatting a File

Menu Utilities->Reformat File... operates on Multiband, Raw, NITF sequential multiband and NITF mono image files. If the file to be reformatted has a *.hdr file these values will be extracted and inserted as defaults into the appropriate edit fields of the Reformat dialog window. A new *.hdr file will be generated along with an updated *.wvl file if one existed for the source image cube. NITF multiband and mono files don't have a separate hdr or wvl file, all sizing information is contained in the source file. Reformatting a NITF file produces a cube or mono file, not another NITF file. Selecting a single output band from a multiband image essentially extracts the band and writes a single <u>raw</u> type image.

The Reformat dialog is shown below.

The upper elements define the spatial format of the data. Normally, this is retrieved from the *.hdr file or directly from a NITF file.

. . . .

This dialog area specifies the format of the converted output file. Its initial values are set equal to the source file size and are edited as desired.

. . . .

Check boxes used to change the order of pixels within a line, the order of the lines within a band and whether any scaling, limiting or byte swapping is to be applied.

The example shown will generate a full size cube scaling 16 bit signed pixels to floating point pixels while limiting the range to [0,1200] and reversing the order of the lines. Swap bytes *will change the byte order to* PC (little endian).

# Pixels:	The number of pixels per line in the source file.
# Lines:	The number of lines in the source file.
# Bands:	The number of bands in the source file.
Header bytes:	The number of bytes in the source file preceeding the image data.
Pixel type:	Size and type of the source pixels.
Interleave:	Structure of the source file: band, line or pixel.

# Pixels, Lines, Bands:	The subset of the source file to be reformatted.
Start pixel, line, band:	Starting coordinates within the source file for subsetting.
Output Pixel type:	Output pixel size. If Limit... is checked then output pixels are kept within the Min & Max range, else they are automatically limited to the range of the pixel type.
Output Interleave:	The format structure of the output cube.
Min, Max:	The lower and upper ranges for limiting the output pixels.
Flip left <-> right:	Change the order of pixels within a line.
Flip top <-> bottom:	Reverse the order of the lines, i.e. first line last.
Apply scaling:	Modify each output band according to the type of scaling selected in Scaling/Calibration. If checked then each band is modified by: scale * band + offset.
Swap bytes...	This check box allows swapping the order of bytes within 16 bit and 32 bit pixels. Note that this applies to the output. The source byte order is taken from the header and if it is unknown you will be prompted for a best guess.
Output header bytes:	Directly copy the source file header bytes to the output file.
Reverse band order:	Output the bands in reverse order.
Reverse wavelength...	Also reverse the order of the output wavelength file.
Scaling/Calibration:	This brings up the following user dialog which is used to determine the type of scaling to be applied to the output. The Apply scaling option checkbox becomes active after returning from this dialog.

Apply this scale and offset to every band.

-- or --

Select a file containing a separate scale & offset pair for each band. The number of pairs must equal the number of bands in the source file.

Scale/Calibrate

Choose type of rescaling:

○ Global – Scale: `1.000`
Offset: `0.000`

◉ Gains/offsets file – [Load...]

[Cancel]   [OK]
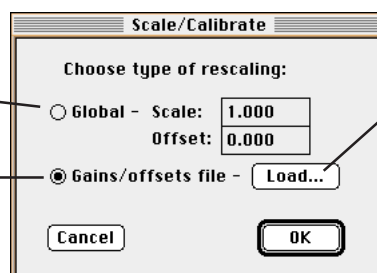
A typical gains and offsets file content: a pair of values for each band...

```
0.00716025  -0.0044436
0.0054349   -1.367508
0.00553983  -1.2478907
0.00583894  -1.2276135
0.00571494  -1.1864042
0.00562544  -1.1921199
0.00553107  -1.5069996
     ...          ...
```

Scale and offset pairs are separated by a space or tab.

Converting a Library to a Cube File

HyperCube can create an image cube whose pixels are the spectral signatures from a library. This is accomplished via Utilities->Library to Image Cube File... which is enabled whenever a Library List window is front most. Each signature is replicated into a square block (for clarity). Blocks are placed side by side and row by row in the same order that the signature names appear in the Library List. The total number of blocks and their rectangular arrangement (per row and the number of rows) is initially chosen by the program and can be edited by you. Null signatures are added to fill out the last row. If you enter values that produce a block total that is less than the number of signatures in the library then only that number of signatures will be converted.

After selecting Library to Image Cube File... the following dialog appears:
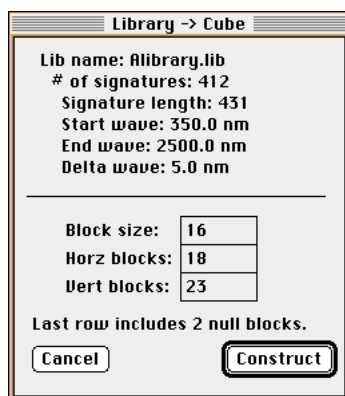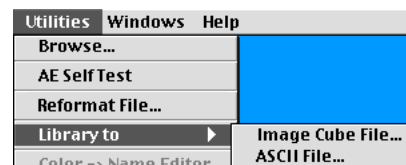
The structure of the library is determined by reading the first spectral signature in the Library List *(see section* Spectral Libraries).

The 431 band image cube will consist of 288 floating point pixels/line by 368 lines with the last 2 blocks containing zero filled signatures.

The upper portion shows the structure of the library while the lower portion defines the size of the resulting image cube. Note in this case that the total number of blocks (18*23) is 2 greater than the number of signatures, 412, in order to make the cube somewhat square and that the resultant image cube file size will be almost 183 megabytes, (16 * 16 * 18 * 23 * 431 * sizeof(float)). These values can be edited and a Block size of 8 (~46 MB) still yields a usable image cube. The image cube is stored in band interleave by pixel (bip) format with floating point values along with newly created header and wavelength files. The Reformat Cube File utility can be used to reconstruct a different format image cube.

If the Library List window already has attached spectral data, as is the case when opened from a Classify window, then you are immediately presented with the save file dialog requesting a name for the image cube. If the library list does not have attached spectral data, i.e. opened using Open As..., Library, then there will be a delay of several seconds while the spectral signatures are being loaded before the save file dialog appears.

The currently active (front most) Library List window can be saved as a text file by choosing Utilities->Library to ASCII File. You are presented with the following dialog describing the size of the library along with several options defining the format to be used when generating the ASCII file.
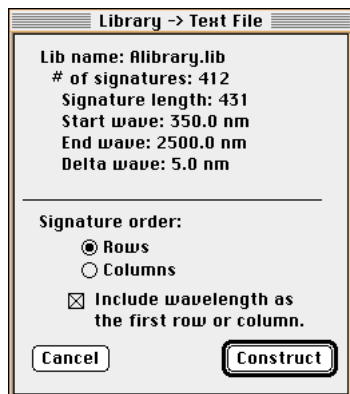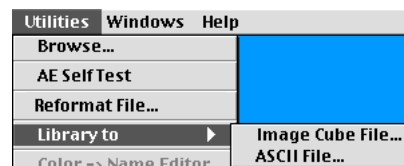
The structure of the library is determined by reading the first spectral signature in the Library List *(see section* Spectral Libraries*).*

If Rows *is selected then each row will be a complete signature (431 points in this case). If* Columns *is selected then a row will consist of all of the spectral values for each signature for a given wavelength (412 values per row shown here).*

Include Wavelengths *adds an extra row or column at the beginning containing the wavelengths.*

If the Library List window is already loaded such as part of a Classify window then the standard save file dialog is presented else there will be a delay while the library data is extracted followed by the save dialog requesting the name of the ASCII file.

LibRow.text

	A	B	C	D	E	F
1	350	355	360	365	370	375
2	0.0224	0.0228	0.0205	0.0208	0.0193	0.0197
3	0.025	0.024	0.022	0.0235	0.0213	0.0208
4	0.026	0.0228	0.024	0.021	0.0216	0.0217
5	0.0239	0.0247	0.0209	0.0238	0.0211	0.0217
6	0.0243	0.0222	0.023	0.022	0.0207	0.0223
7	0.0302	0.0272	0.0263	0.0276	0.0259	0.0285
8	0.0201	0.0168	0.0172	0.016	0.0158	0.0173
9	0.0202	0.0176	0.017	0.0158	0.016	0.0166
10	0.0178	0.0192	0.016	0.0197	0.0158	0.0179
11	0.0308	0.03	0.0262	0.0294	0.0253	0.0266
12	0.0315	0.0291	0.0279	0.0263	0.0266	0.0255
13	0.033	0.0294	0.0283	0.0279	0.028	0.0256
14	0.0207	0.0171	0.0177	0.0167	0.0158	0.0176
15	0.02	0.0206	0.0196	0.0192	0.0183	0.0189
16	0.0301	0.0263	0.028	0.0251	0.0256	0.0244

Small portions of a library saved in row and column form respectively and opened in Excel™. In each case the wavelength option was included.

LibColumn.text

	A	B	C	D	E	F
1	350	0.0224	0.025	0.026	0.0239	0.0243
2	355	0.0228	0.024	0.0228	0.0247	0.0222
3	360	0.0205	0.022	0.024	0.0209	0.023
4	365	0.0208	0.0235	0.021	0.0238	0.022
5	370	0.0193	0.0213	0.0216	0.0211	0.0207
6	375	0.0197	0.0208	0.0217	0.0217	0.0223
7	380	0.0188	0.0211	0.0203	0.0221	0.0197
8	385	0.0173	0.0198	0.0209	0.0212	0.021
9	390	0.0188	0.021	0.0192	0.0232	0.0203
10	395	0.0171	0.0194	0.02	0.0221	0.02
11	400	0.018	0.0206	0.0204	0.0238	0.0208
12	405	0.0186	0.0217	0.0222	0.0252	0.0214
13	410	0.0202	0.0234	0.0236	0.0273	0.0235
14	415	0.022	0.0249	0.025	0.0295	0.0244
15	420	0.0229	0.0267	0.0271	0.0311	0.026
16	425	0.0251	0.0298	0.0288	0.0341	0.0284

Concatenate Images

Two images may be combined side to side or top to bottom to form a new single image by choosing Utilities->Concatenate Images. The images don't have to be the same spatial size, the output will be sized to the minimum dimension of the two. They must have the same pixel depth, i.e. both 8 bit or both 32 bit.

The dialog below shows the result of combining two bands from an image cube that were first separated from the cube using Duplicate in the Edit menu.
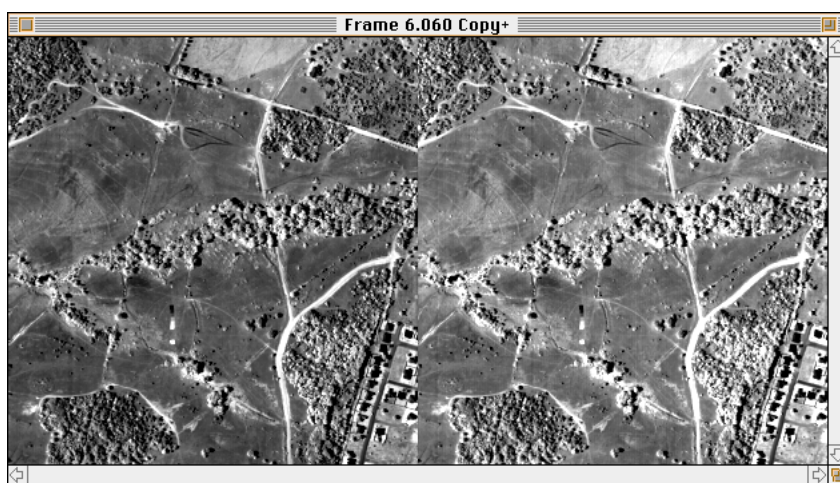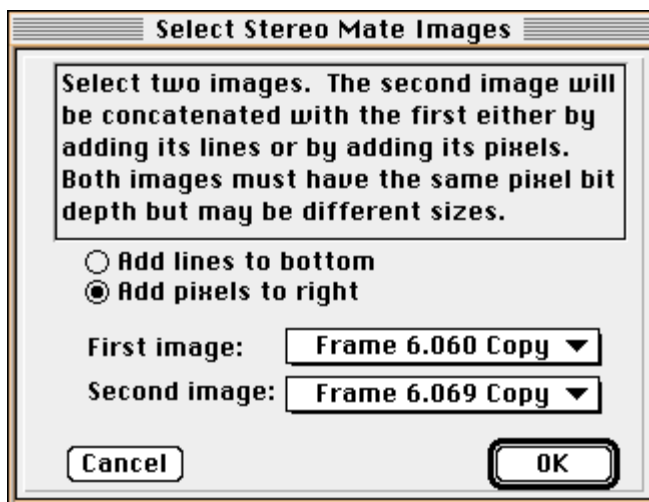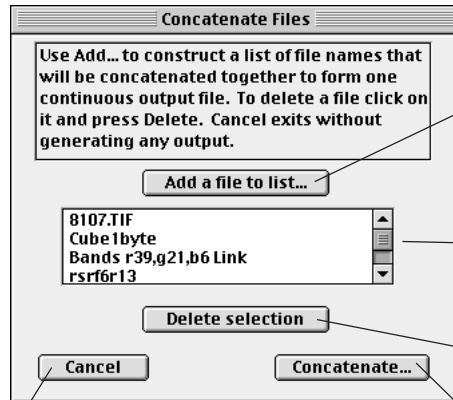

Bands 60 and 69 from an image cube concatenated left to right. The bands were first extracted from a loaded cube via Duplicate.

Concatenate Files

This utility allows you to make one continuous file from two or more files by reading each file and sequentially concatenating their contents to create a single new output file, e.g. combining individual Landsat file bands into a contiguous BSQ (Band Sequential) cube. The source files are not altered in any way. Compare this with the previously described Concatentate Images, which operates only on images loaded for display.

You start the process by selecting menu Utilities->Concatenate Files which brings up the following dialog:

Each button press displays the standard open file dialog. Any type of file may be selected.

Each selected file is added to the list. A selection may be removed by clicking it and pressing Delete Selection.

Ignore all selections and exit without creating an output file.

Display the standard save file dialog and solicit a name for the concatenated file.

Changing a Window Title

It is often convenient to be able to change the title (name) of a window without having to first do a Save as... . An example is when printing where the title can be changed to something more descriptive and then listed below the output (see section Printing and note below). To perform the change activate the window and choose Utilities->Change Window Title. This brings up the following dialog.
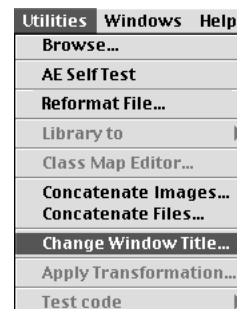
If checked and window is a cube then the loaded bands will be renumbered starting at 1.

The current window name, Frames 5-8r
being changed to a more descriptive title.

Note: Window titles may consist of up to 63 characters, while file names are limited to 31 characters. You will be alerted when saving an image window with too many characters.

Apply Transformation

The active image cube file can be transformed into another cube file by mulitplying each spectral vector (signature) by a linear transformation in the form of an M x N matrix. The matrix must be an ASCII file (File->Save As Text and and answering affirmative to Matrix) or such as those optionally saved via menu items Image->Convert Image->Principal Components, Functions->Statistics->Band Correlations and Functions->F(Signatures). The format consists of a single line containing the dimensions of the matrix, (# rows <u>tab</u> # columns) followed by the M rows of the matrix with each of the N column elements separated by a space or tab and ended with a return. An example of a 4x3 transformation matrix is:

```
4    3
4.6864242694e-01    -6.9847956895e-01    2.5492193504e-01
5.0310142892e-01    4.5755700310e-01    7.0505069625e-01
5.0999322932e-01    -2.8597312694e-01    -3.8238290256e-01
5.1688502971e-01    4.7009281141e-01    -5.4009410160e-01
```

Algebraically, the following occurs:

$AP = B$ for the forward transformation

$AP^{-1} = B$ for the inverse transformation

$AP^T = B$ for the transpose transformation, where

A is each 1 by M spectral signature in the source cube

P is the M by N transformation matrix

B is each resulting 1 by N new spectral signature

P^{-1} is the inverse of matrix P, P^T is the transpose of P

If the Output complement option is checked then A - B is formed.

You can use less than M rows or N columns of P, the unused rows are treated as if they are zero and unused columns reduce the output vector dimension to 1 x cols. This allows you to construct an output cube using just the first significant vectors (columns) of a matrix file (see sections Image to Principal Components and F(Signatures)).
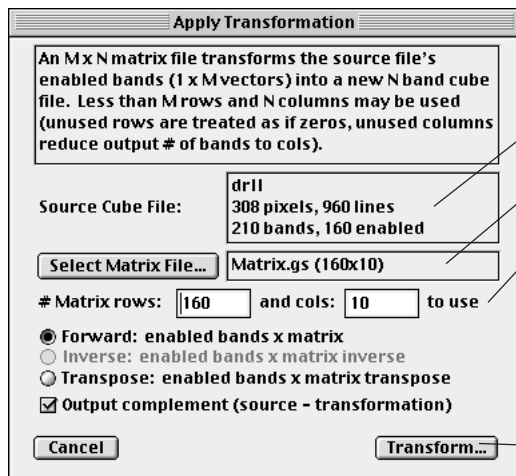
The source cube that will be transformed.

The transformation matrix to be applied.

The number of rows and columns of the transformation matrix to use. In this case all of the rows and columns will be applied. Setting cols *less than the number of enabled bands reduces the output cube to the same depth as* cols.

Generate the new transformed cube file. It will be in BIL *floating point format.*
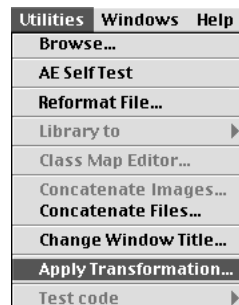
Applications

Shaded Relief

An image can be rendered into a synthetic image that depicts the effect of uniformly illuminating it with a parallel light source, e.g. the Sun. Each gray value, or its associated (attached) data (see sections Raw, Multiband, NITF Files, TIFF Files), is interpreted as an elevation that intersects and reflects the light source as a function of the angle of incidence between the light and the slope of the elevation. Choosing Applications->Shaded Relief... when a gray image is the front most window brings up the Shaded Relief dialog. Any excluded pixel values are ignored when computing the initial x and y slope scales (see section Pixel Exclusion).

Elevation angle above the data plane of the light source.

Heading angle measured CCW from the east or positive X direction.

The denominator used in computing the slope in the X and Y directions. Initial values are computed from the average slope within the image. The lesser these values the greater the shadowing effect.

Shaded Relief

Sun elevation is measured from the horizon and azimuth is CCW from the X (East) direction. If Anaglyph is checked, Parallax is used to contruct a matching left, right stereo pair and red, cyan image. Parallax is in pixels.

Elevation: 45.0   Azimuth: 135.0

X slope scale: 40   Y slope scale: 40

☐ Anaglyph, Parallax: N/A

[Cancel]   [OK]

If Anaglyph is checked then a red:cyan stereo anaglyphic image is also generated using the specified Parallax value. See section Stereo Mate for an example.

Gray image representation of data used to generate the Shaded Relief image shown to the right using the values defined in the above dialog.
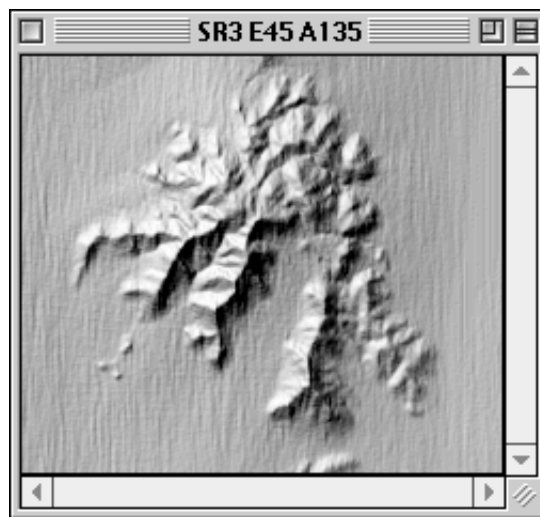
Resulting Shaded Relief with the light coming from the upper left 45 degrees above the "horizon". The fine vertical striping is part of the data and not an artifact.

Stereo Mate

A Base image along with a Data image representing elevation that is spatially registered to the Base image can be used to form a new version of the Base image with elevation induced displacements. If the Base image is 8 bits and is used as the red component and the Stereo Mate as the green and blue image components. The resulting color anaglyph can be viewed in stereo with red and cyan glasses. If the Base image is color the Stereo Mate will also be color and if placed along side of each other as narrow strips you can view with a pocket stereoscope.
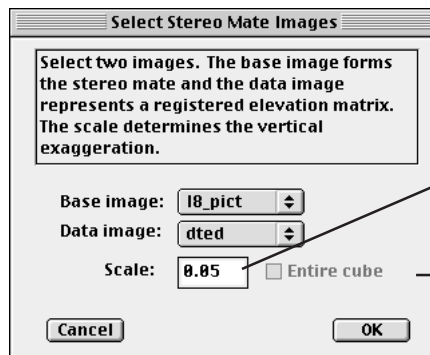
Multiplicative factor applied to the Data *image.*

If the Base image is a cube band then this check box allows generating a stereo mate for every band in the cube.
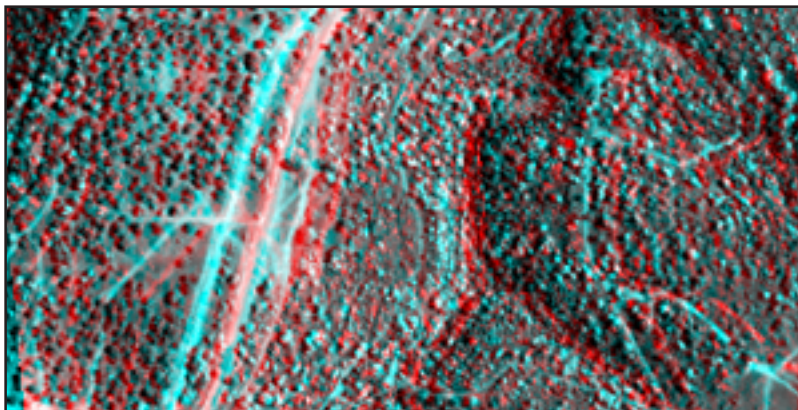
An anaglyphic stereo image generated by using R,G,B to Color *and selecting red as the base image and blue and green as the stereo mate. View via red and cyan glasses.*

Perspective

A Data image of elevations can be reconstructed as if viewed from almost any perspective by applying basic photogrammetry. If there is a corresponding Base image, i.e., each feature pixel is one-to-one with an elevation then the features themselves can be drapped across the elevations to yield a different perspective. The process is initiated by selecting Perspective... from the Applications menu which displays the following dialog. All values may be interactively modified.
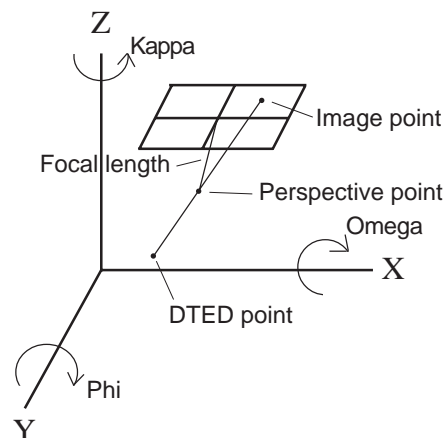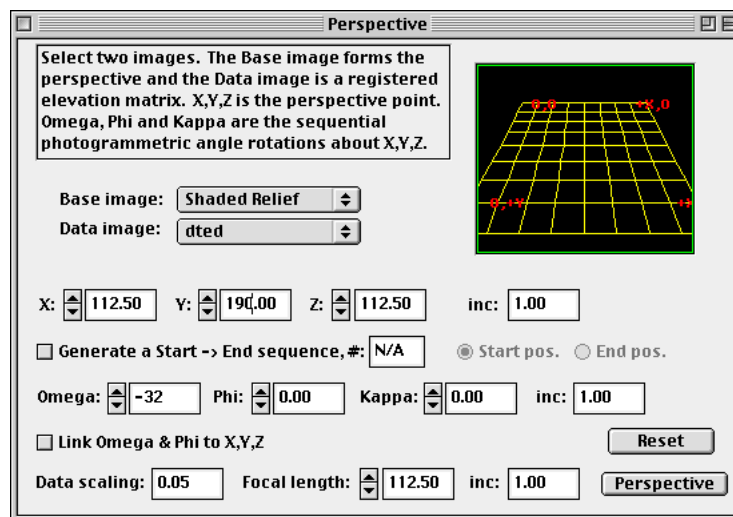
Base Image:	Feature image that is transformed into the perspective view. The Data image may also be used if no Base image exits.
Data image:	An image whose pixels are interpreted as elevations. See section Raw.
Starting pos.:	The coordinates of the perspective or view point relative to the Base image. The origin is the upper left pixel with positive X in the right direction and positive Y moving downward. Z is above the image. After selecting the Base image, X,Y,Z are preset to above the middle of the Base image at a height of the maximum of X and Y. You can readily edit these values. Note that X,Y,Z is a point, not a direction, that is determined by Omega, Phi and Kappa.
Ending pos.:	If Generate a sequence, below, is checked then this is the ending view point of a uniform set of view points and angles that will be used to generate a synthetic cube movie.
Generate a sequence:	If checked then # equally spaced view points are constructed and assembled into a cube.
Omega, Phi, Kappa:	The rotations about the X, Y, and Z axes respectively in degrees. If Link Omega & Phi is checked then the perspective orientation is fixed pointing to the center of the Base image. Therefore, a Sequence will produce a fly-by perspective with an invariant center.

Link Omega & Phi to X,Y,Z: If checked then Omega and Phi are automatically computed as a function of X and Y so that the perspective view is pointed at the center of the Base image. You can check this box to force a centered view, click an Outline button and then uncheck it to modify Omega and Phi manually for a more desirable view. See the Outline buttons described below.

Data scaling: The multiplicative factor applied to the Data. The smaller the number the "flatter" the perspective appears. Usually, a value of .05 is a good starting point. The larger the scaling the more exaggerated the effect of elevation and the longer the computation time.

Focal length: How far behind the perspective point the imaging plane lies. The greater the focal length the larger the nominal magnification. A small Focal length is analogous to a wide angle lens. The initial value is set equal to the Z coordinate and produces a one-to-one scale.
Changing the Focal length does not change the output image dimensions.

Reset: Set the perspective back to the initial starting values as if viewing the image from a point directly over the center with no rotations.

Perspective: Generate the perspective image(s).
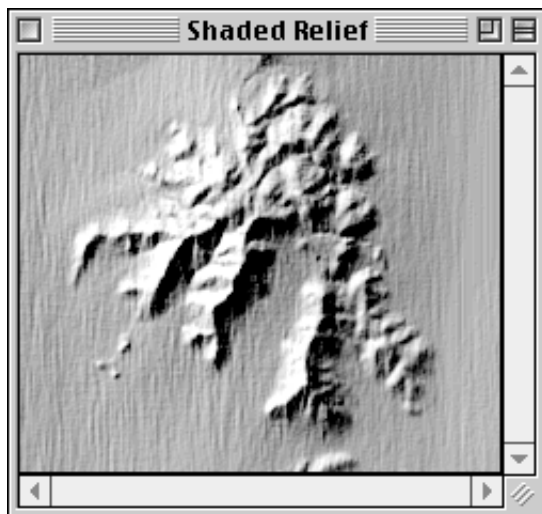
A shaded relief image generated from elevation data is used as the Base *image and provides the features that will be viewed from a different perspective as shown to the right.*
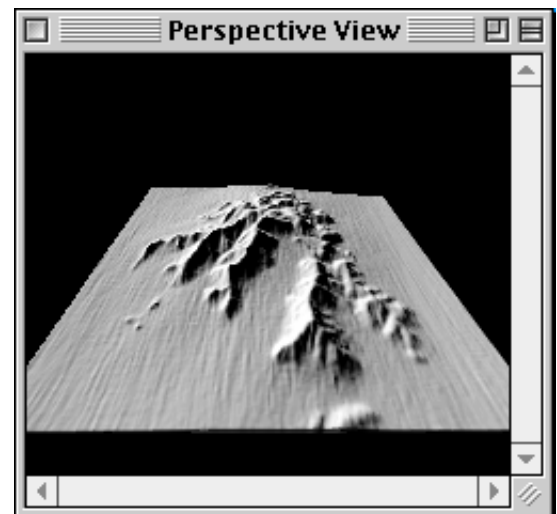
The reconstructed perspective image using the photogrammetric parameters entered into the previously described user dialog.
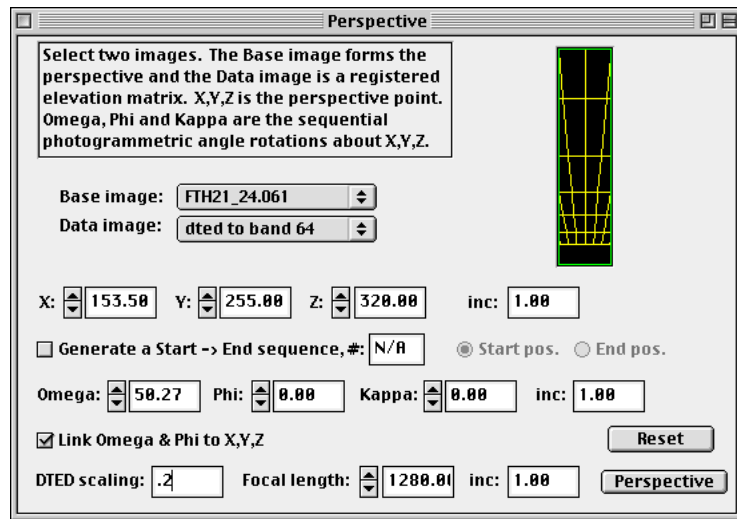
The above dialog was used to select and generate the perspective image shown here. Both images were then rotated 180 degrees to produce the final views shown below. Note that Link *has been selected to force pointing at the center of the source image.*
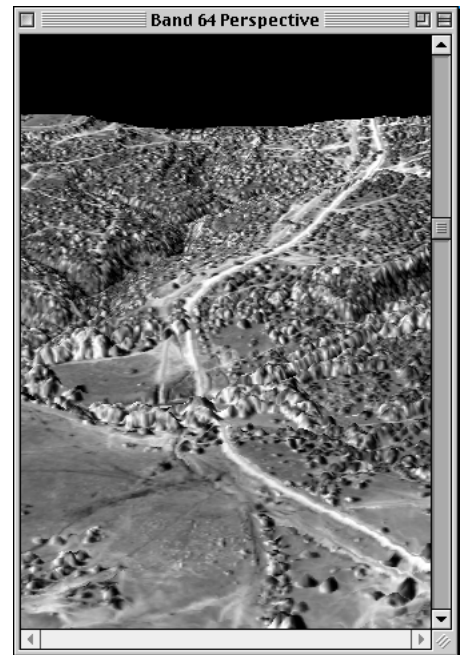
Radiance

This application processes thermal images, generally in the range of 3 to 18 microns, that are in radiance units of $Wm^{-2}\mu m^{-1}$. Structurally, there is no distinction between a thermal image cube and a visible to SWIR cube, only the wavelength definition. HyperCube allows you to generate a temperature data <u>image</u> and optionally, an emissivity <u>cube</u>. The temperature is derived by comparing the radiance of each image cube pixel with the radiance of a series of black body curves. The closest match, via a user specified tolerance, produces the temperature for that pixel. There is also a utility, Plot BB Curve, that can display a black body radiance curve for any given temperature.

Shown below are the various user dialogs that are available. Note that the routines require wavelengths to be in microns (μm) but that the temperature can be in centigrade, Kelvin or Fahrenheit. Any questions regarding the applicability of these functions should be directed to **evans@tec.army.mil**.
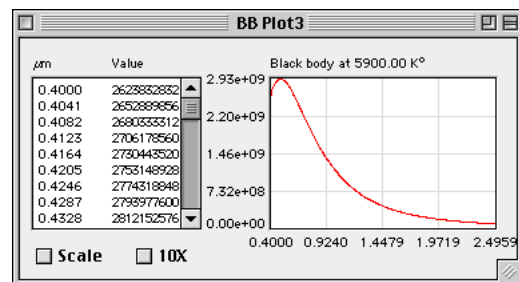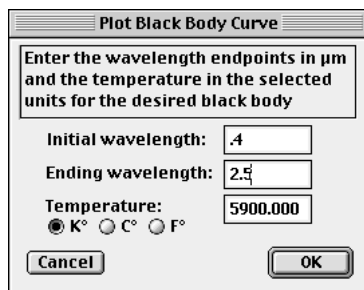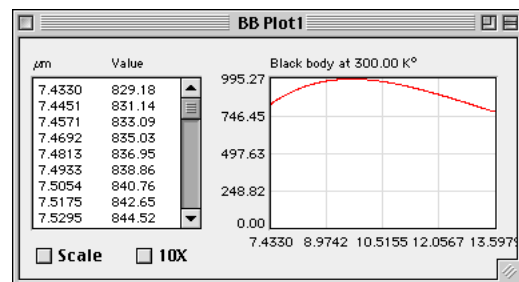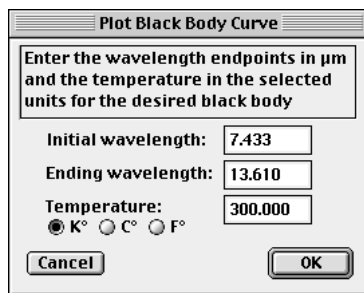
Applications->Radiance->Plot BB Curve *dialogs, on the left, and their corresponding black body radiance curves on the right for the given temperature and wavelength ranges.*

Menu Applications->Radiance->Emissivity displays the following dialogs. The default values shown were used to produce the displayed outputs from a thermal image cube.
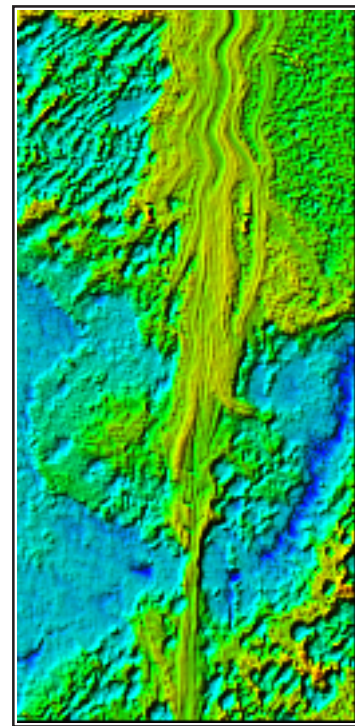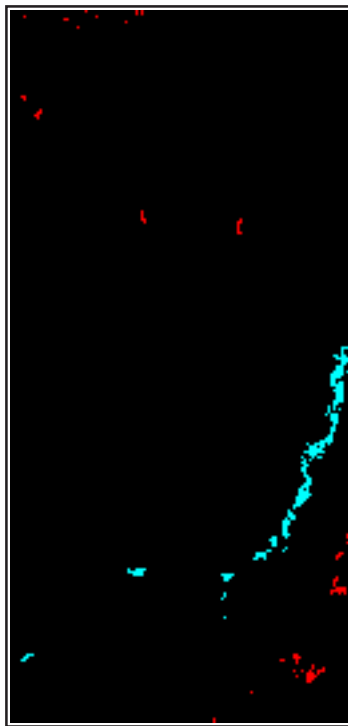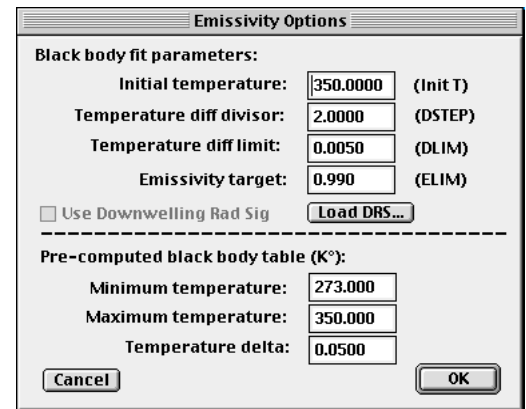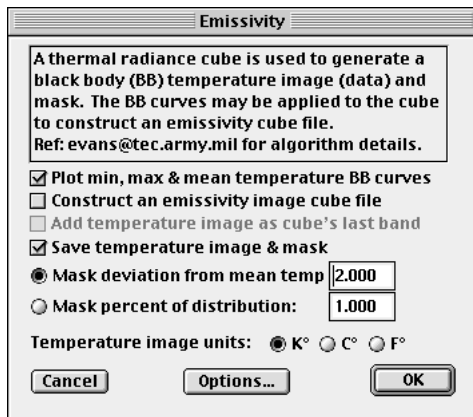
The output image at left is a mask showing the warmest, red, and coolest, blue, regions within the thermal image cube, i.e., those temperatures 2 or more deviations from the mean. At right is a shaded relief of the temperature image that has been similarly coded. The coolest region coincides with a creek. Below is the histogram of the temperature data image in units of degrees Kelvin.
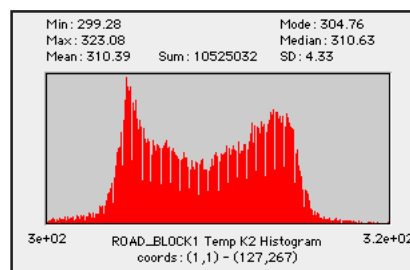
Miscellaneous

Icons

HyperCube has a family of 8 Finder icons. Five of these are regular Mac data types and represent TEXT, JPEG, PICT, TIFF and Sun Raster Format files. The other three represent the HyperCube application, unformatted binary data and PhotoShop™ convolution kernels as shown in the figures below.

HyperCube application icon.

ASCII text file, e.g., *.hdr, *.wvl.

Image files saved in JPEG, NITF, Pict, SRF, and TIFF formats.

Unformatted image file saved as Raw.

PhotoShop™ convolution kernel of type 8BCK.

Double clicking an image icon will automatically launch HyperCube and the load the corresponding image. Similarly, a *.wvl or *.hdr icon can be double clicked as well.

The Windows version does not have an icon family, only the application has an icon. Any HyperCube documents must be opened within the application.

Error Messages

When HyperCube detects an internal error such as running out of memory it generally can recover enough to allow you to at least save and print. There is one serious type of error dialog that may be presented indicating that an immediate exit is warranted. But if you have generated some result that just must be saved or printed then HyperCube can be forced to continue at

risk of an immediate crash. Generally, the exit path should be taken especially if it is followed by another dire warning.

This error message will not appear in the Windows version.
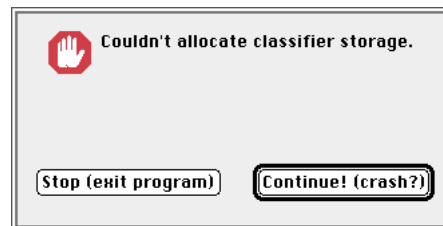
Capturing the Screen

The contents of the main display screen may be captured at any time other than a modal dialog (i.e. those demanding attention) by simultaneously pressing the command 1(one) keys (Ctrl 1 in the <u>Windows</u> version). This creates a new true color window of the entire display that may be saved like any other pict image. Note that a full screen color window requires approximately 2MB for 832 pixels by 624 lines. HyperCube checks for sufficient memory and warns if it is insufficient.

Additionally, in the <u>Windows</u> version you can automatically copy the entire active window to the clipboard by depressing Ctrl-Print Screen. Then choose File->New... followed by Edit->Paste to place it in a window where it can be saved.

Saving and Copying Scrollable Lists

Any window which has a scrollable list, e.g. spectral plot, band correlations, that does not contain a specific Save button may still be written to a file by selecting Save from the Edit menu. This will bring up the standard file save dialog. The data will be formatted as tab delimited ASCII text and can be reread by any application that reads text. Also, this same scrollable list may be copied to the clipboard as tab delimited, c/r terminated lines of text by choosing the Edit->Copy menu and then <u>pasted</u> into another application.

Loading/Saving Text Files

Any small text file can be opened and displayed in a scrollable list. This is especially useful for examining header files. The file is opened by choosing menu File->Open As and Text from the fromat dialog popup. It can then be edited (Cut, Copy and Paste) and resaved. The up and down arrow keys and the page up, page down keys are used to navigate through the text. Choosing File->New... and selecting the Text option will open a blank Text window that may be typed into and saved as text.

An image or any selection may be saved as an ASCII text file by choosing menu File->Save As and Text as the format (and optionally as a matrix, see section Apply Transformation). The pixels are tab delimited with each line followed by a return. The size of the resulting file may be 4 or 5 times larger than the source.

Note: Any embedded zeros within a text file are automatically removed when the file is read. Carriage returns on the <u>Windows</u> version are followed by a newline character.

```
dma_dted.hdr

samples = 512
lines = 512
header offset = 0
file type = dted
data type = 2
xstart = 1
ystart = 1
numpixels = 512
numlines = 512
byte order = big endian
autoscaling = yes
percent = 100
```
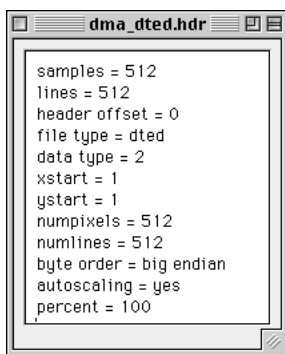
Using menu File->Open As, Text *and selecting a* DTED/Data *header file results in the window at left. This may be edited and saved, functioning as a simple word processor. Dragging the cursor within the window's content area will automatically scroll exposing additional lines as will the arrow and page keys. Text will either wrap-around or limit itself to a line if a carriage return occurs within the width of the window. The window may be resized as necessary.*

HyperCube Menus

Edit	Applications
Undo	⌘Z
Cut	⌘X
Copy	⌘C
Paste	⌘V
Clear	
Select All	⌘A
Duplicate	⌘D
Options	▶

Image	Functions	Utilities
Convert Image		▶
Add to Cube...		
Attach Data...		
Zoom In		⌘+
Zoom Out		⌘-
Flip		▶
Rotate		▶
Shift		▶
Color Map...		
Gray Map...		
Filter		▶
Magnify Image		▶
Embed Wavelength		
Cube Color Composite		▶

Functions	Utilities	Wind
Flicker Cube		
Flicker/Superimpose...		
Statistics		▶
Plot		▶
Arithmetic...		
Overlay Color...		
Adjacency		▶
Classify...		
Calibration...		
F(Signatures)...		
Fourier...		
Mosaic...		
Warp...		

Utilities	Windows	Help
Browse...		
AE Self Test		
Reformat File...		
Library to		▶
Class Map Editor...		
Concatenate Images...		
Concatenate Files...		
Change Window Title...		
Apply Transformation...		
Test code		▶

Windows	Help
Show Info	
Show Band List	
✓ g1	
Tulips	
Color cube	
Bands r3,g2,b1 Link	
Bands r3,g2,b1 Link Copy	

Browse *and* AE Self Test *are in* <u>Mac</u> *version only.*

File	Edit	Applications
New...		⌘N
Open...		⌘O
Open As...		
Load Selection...		⌘L
Close		⌘W
Close All		
Save		⌘S
Save As...		
Save Bands...		
Page Setup...		
Print...		⌘P
Quit		⌘Q

Applications	Imag
Shaded Relief...	
Stereo Mate...	
Perspective...	
Radiance	▶

See Table of Contents *section* Applications *for details.*

The Windows *menu will contain* Cascade *and* Tile *in the* <u>Windows</u> *version but not the* <u>Mac</u> *version.*

Open As... *is a* <u>Mac</u> *only menu item.*

The availability of a specific menu function is directly keyed to the current active window. If an item is not enabled and you think it should be, it's most likely because it is not applicable to the front most window or is ambiguous. For example, Embed Wavelength is enabled only when a cube window is front, likewise for Classify. Fourier and other operators can only be used when an image window is active and Convert Image is very dependent upon the active window's bit depth.

Some menu functions produce operators that become attached to the window that is active. Color Map, Gray Map, Filter, Flicker Cube, and Fourier are examples of attached operators. When the parent window is closed the operator window disappears.

Although Browse and AE Self Test are enabled they are related to future inter-application capabilities.
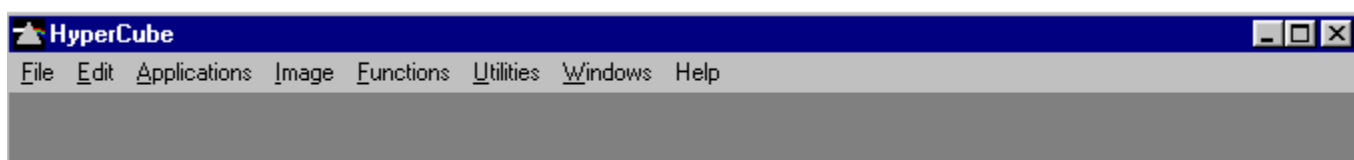
<u>Windows</u> *version main* Menu Bar.

New...   Create a new empty image window that can be used to paste the current clipboard. You are presented with the following dialog options which are preset according to the contents of the clipboard.
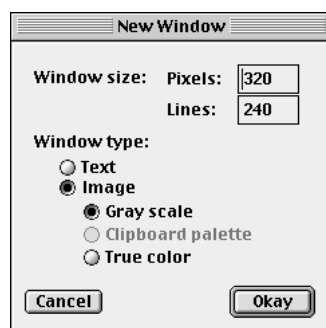
Pixels	Width of window
Lines	Height of window
Text	Create a new Text window
Image	Create a new Image window.
Gray scale	8 bit gray window
Clipboard...	8 bit pseudo color window
True color	32 bit full color window

Open...   Presents the standard Mac open file dialog showing only Pict type files.

Open As...   A modified open file dialog which includes TIFF and other image files.

Load Sel...   Available when an overview is front most. Brings up the last dialog used to open the source image with the requested size fields set to correspond to any selection rectangle that appears on the overview. See section Overviews.

Close   Close and dismiss the currently front most window. Any modified window first presents an opportunity to save it.

Close All   Closes all open windows.

Save   Save the current active image window as a Pict file.

Save As...   Present a modified save dialog allowing you to save in Pict and other type formats.

Save Bands...   Sequentially save each band of a cube file window as individual Pict files.

Page Setup...   Bring up the standard printer setup dialog for the chosen printer.

Print...   Display the print dialog and print the contents of the front window.

Quit   Immediately exit HyperCube. No warning is given to save anything.

Undo   Not implemented in the present HyperCube version.

Cut   Also not implemented as yet.

Copy   Copy the current front window to the clipboard. If a selection rectangle is present then copy just its contents.

Paste   Paste the contents of the clipboard to the front window. You can position it before embedding.

Clear   De-toggles enabled items in Library list, Classify points list and Cube points.

Select All   Toggles on all items in a Library list or Classify points list.

Duplicate   Make an exact copy of the current front window contents but not its functionality. Equivalent to performing: Copy, New and Paste.

Options A submenu with the following sub-functions.
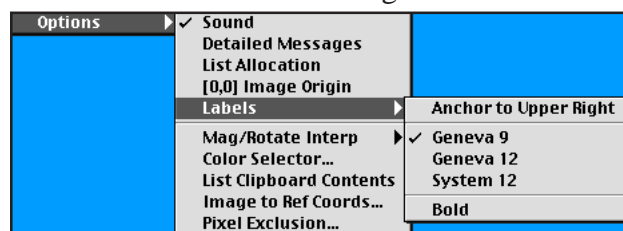
Sound	Toggle alert sound on/off.
Detailed Mess...	Toggle debug info on/off.
List allocat...	Display a snapshot of internal structures.
[0,0] Image Origin	Change the upper left coordinate origin from [1,1] to [0,0]
Labels	A submenu specifying the font, size and position of dynamic labels (see Dynamic Displays and Plots).
Mag/Rot Interp...	Interpolation for magnify and rotate (see Image menu).

Color Selector...	Display the selection point to color correspondence editing window.
List Clipboard...	What's currently in the clipboard.
Image to Ref...	Display a dialog defining the correspondence between an image coordinate and a reference coordinate.
Pixel Ex...	Displays a dialog specifying certain pixel values that are excluded from histogram computations. See section Pixel Exclusion.

Convert Image A submenu used to change the structure of an image.
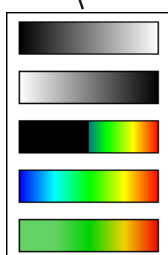
R,G,B to Color...	Select 8 bit images to form a color image.
H,S,I to Color...	Select 8 bit hue, saturation and intensity to form a color image.
Indexed to Color	Changes 8 bit pseudo color to true color.
Complement	Complement a color image.
Color to R,G,B	Separate a color image into planes.
Color to H,S,I	Separate a color image into hue, saturation and intensity.
Color to Indexed	Change true color to pseudo color.
Replace Color...	Substitute one color with another.
Image to Mask...	Change a specific color (gray) to a mask.
Grayscale	Convert color to gray image.
Hist Equalize	Construct histogram equalize image.
Hist Equal ea.	As above, but each color separately.
Principal Comp...	Convert color to principal components.
Plane to Cube	Use a gray image as cube base. This menu item toggles with Color to Cube.
Color to Cube	Convert true color image into a 3 band image cube.
New Cube Link...	Change the present cube color composite link.
Color Table	Change present 8 bit pseudo color table.

Attach Data... Retained data from one image is copied and attached to another image.

Add to Cube... Presents a dialog list from which to choose images for a new cube.

Zoom in Magnify the current image window by 2.

Zoom out Demagnify the current image window by 2.

Flip A submenu allowing images and cubes to be reversed and reflected.
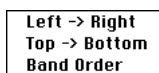
Left -> Right	Reverse the pixels within a line.
Top -> Bottom	Reverse the order of the lines.
Band Order	Change the loaded bands within a cube so that the last is first.

Rotate Multiple 90 degree rotations of image and cube windows.
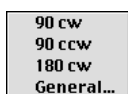
90 cw	Rotate image or cube 90 degrees clockwise.
90 ccw	90 degrees counter clockwise.
180 cw	180 degree rotation.
General	User input of rotation.

Shift   Left and right pixel and line cyclic shifts of image windows.

Left	Shift all pixels or only even pixels right 1.
Right	Shift right either all or only even.
Up	Move all lines up 1 line.
Down	Move all lines down 1 line.

Color Map...   Display interactive dialog of color balance controls for color images.

Gray Map...   Change the contrast and brightness of any image or cube window.

Filter   A submenu of static and interactive spatial domain image filtering.
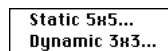
Static 5x5...   Display a static 5 by 5 kernel convolution static dialog.

Dynamic 3x3...   Display an interactive 3x3 kernel convolution dialog.

Magnify ...   Toggles between **Magnify Image** and **Magnify Selection** depending on whether the active image window has a selection rectangle. Generates a new magnified/minified view of the image or selection rectangle by the factor chosen in the submenu.

Embed Wave... Write the wavelength into each band of the loaded cube. Useful before Save Bands function.

Cube Color Composite   A submenu enabled when a cube image is front. Generates a true color image using the values set in Definitions... to determine which bands of the cube file to use. The new color image is automatically linked to the cube (see sections Color Composite and Classify Functions).
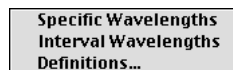
Specific Wave   Use the 3 closest bands to the values set in Definitions...

Interval Wave   Integrate all bands lying in each interval to form the red, green and blue components.

Definitions...   Display a dialog allowing you to specify the red, green and blue wavelengths for a cube color composite.

Flicker Cube   Generate a new cube window devoted to flickering between bands.

Flicker/Superimpose Display a dialog of image names from which to choose the sources for an image flicker or image superposition.

Statistics   A submenu of statistical functions to be applied to the current cube.

| Pair Correlations... |
| Band Correlations... |

                   Pair Corr...   Cross correlation computation between 2 images.
                   Band Corr... Cross correlation between successive cube bands.

Plot   A submenu of graphical operators.

| Histogram |
| Spectra (Selected Points) |
| Spectra (Dynamic) |
| Scatter... |
| Profile |
| Scaling... |
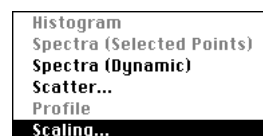
    Histogram   Perform a pixel distribution histogram of the current image or cube.

    Spectra (Se...   Display the spectral plot window for the selected active cube points.

    Spectra (Dy...   Dynamically display a spectral plot as the cursor moves over a cube.

    Scatter...   Generate a 2 dimensional plot of the pixel distribution between 2 images.

    Profile   Draw a gray level density plot of the active image profile line. If the image is color draw plots for red, green and blue.

    Scaling...   Selectable when a plot window is active. Allows you to change the plot vertical axis.

Arithmetic...   Open an interactive window allowing you to execute arithmetic expressions involving 8 bit images.

Overlay Color... Generate a composite image from a mask and base image.

Adjacency...   A submenu for comparing and outputting spatially near pixels.

Classify...   Open up the classification dialog window for the current image cube.

Calibration...   Open the calibration dialog for the frontmost Classify window.
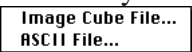
F(Signatures)... Compute eigenvector matrix of Classify signatures.

Fourier...   Display an interactive dialog to perform frequency domain filtering upon the current 8 bit image window.

Mosaic...   Display an interactive dialog to select and control the merging of two or more images into a composite assemblage.

Warp...   Display an interactive dialog to select and control the geometric transformation of one image to another.

Browse... — Presents a list of running applications on the current network. For future distributed processing.

AE Self Test — HyperCube sends an Apple Event message to itself. If Detailed Messages (Edit menu) is toggled on then the details of this message are shown. Not in <u>Windows</u> version.

Reformat File... — Open a dialog for reformatting the spatial and/or pixel type of a file.

Library to — A submenu to convert the current front most library list window into an image cube file or an ASCII text file.

> Image Cube File...
> ASCII File...

Class Map Editor — Enabled when a classification image is front most. Permits changing the region color and name association.

Concatenate Images... — Display a dialog allowing you to select two images that will be concatenated in either pixel or line order to form one new image.

Concatenate Files... — Display a dialog allowing you to select multiples files to be concatenated into a single sequential new file.

Change Window Title... — Solicit a new title for the current active window.

Apply Transformation... — Multiply a cube file by an orthogonal transformation.

Windows

Show Info — Display a small dynamic window that shows various HyperCube details as the cursor passes over other windows.

Show Band List — Display a listing of the wavelengths of the current image cube and allow you to set the status of various bands.

Cascade — <u>Windows</u> version only. Arranges all of the open windows in an overlapped diagonal order.

Tile — <u>Windows</u> version only. Resizes and rearranges the open windows in a row by row pattern.

"Names" — A dynamic list in order of creation of HyperCube windows. The front most window is preceded by a check mark.

Index

Symbols

% 14, 24, 25
*.c2n file 103
*.hdr 12, 110, 112, 127
*.wav 110, 112, 127
.ev 94
.gs 94
.pm 94
.wav 23, 64
.wvl 23, 64, 110, 112, 127
[0,0] 34
[0,1] 15, 71
[1,1] 34
{constant,constant,constant} 49
180 cw 132
8BCK 127
90 ccw 132
90 cw 132

A

Abs diff 78
Add to Cube... 132
Adjacency 52
AE Self Test 129, 135
Affine 63
  image to reference 22
Algorithms 78
Anaglyph 120, 121
Arithmetic 91
  sequence 49
Arithmetic operators 48
Arithmetic... 134
ASCII 60
  Saving images 128
Asterisk 23
Attach Data 41
Averaging
  spectral plots 76
Azimuth 120

B

Band Correlations 43, 134
  sequential 43
  specific 43
Band List 78, 107
Band list
  revert 65
Band Order 30, 132

Band Scrolling 66
Binary 39
Black body 125
Block size 114
Brightness 27
Browse... 129, 135
Byte 109
Byte order 13, 112

C

Calibration 106
Cascade 135
Class Map Editor
  ESRI Shape Files 7
  singularities 55
Classic 5
  Info window 34
  Pict 6
Classifications
  principal components 36
Classify 54, 55
Classify Each 85
Classify Mean 85, 96
Classify... 78, 129, 134
  overlays 91
Classmap Editor
  Color to Classmap 41
Clear 26, 71, 131
Clear Lib 71
Close 130
Close All 130
Coefficients 25
Color Balance 29
Color Composite 83, 87, 133
Color Image 88
Color map
  histogram 44
Color Map... 29, 129, 133
Color Mask 88
Color overlay 38
Color Selector 85, 87, 100
Color separation 39
Color Table 132
Color to cube 69
Color to Indexed 37, 132
Color to Library 100
Color to Planes 35, 132
Columns 119
Command-option-C 38, 73

H

Harmonic 78
HDF
  Hierarchical Data Format 8
Header 12
  *.hdr files 19
  saving 14
Header file
  image to reference 22
Histogram
  color mapping 29
  exclusions 45
  file 44
  gray mapping 29
  of classifications 90
  Plot 134
  window 44
Histogram equalization 40, 70
Hit rate 96
Hotelling 36
H,S,I 35
Hue... 35

I

Icons
  appl 127
  Pict 127
  Raw 127
  Text 127
icords
  keyword 21
Image Arithmetic 48
  ** operator in 48
  abs,...,atan2 48
  remapping 49
  shift_left... 48
Image cube
  adding to an 69
  constructing an 69
Image to Mask 39
Image to Reference 105
  change in origin 22
Indexed to Color 37
Info
  band statistics 43
Info Window 34, 71
  cursor position 34
  NITF coordinates 10
  Show Info 34
  spectral coords 71
Interleave 15, 109, 114
Interpolation 63
Interval Wavelengths 70

J

JPEG 10, 11, 37
  baseline 9
  compression 9
  markers 9

K

Keywords
  definitions 20
Kmeans 78, 82
  class color 82
  classes 82
  iterations 82
  options 82
  plots 82

L

Latitude 59
latitude 21
Left 31
Left -> Right 30, 132
Lib Key 100
Lib:File
  match lists 100
Library to Cube 114
Library... 84, 99, 114
  List 114
Linear 27
  correlation 42
Link 41
Link to Cube 35
Linked image 83, 133
List allocation 131
Loading
  by double clicking 64
Longitude 59
longitude 21
LZW 11

M

Mac version 8.6 5
Mag Interpolation 131
Magnify
  bilinear 33
  cubic 33
  minification 133
  replicate 33
  selection rectangle 33
Magnify Image 133
Magnify Selection 133